

sinclair
ZX Spectrum
Voss
ZX
Spectrum
El manual escolar
UN LIBRO DATA BECKER
EDITADO POR FERRE MORET, S.A.

089-6085
020-2300


sinclair
ZX Spectrum
CAPS SHIFT
ENTER
SPACE
Voss
ZX
Spectrum
El manual escolar
UN LIBRO DATA BECKER
EDITADO POR FERRE MORET, S.A.

Los programas y utilidades que se encuentran en este libro pueden ser adquiridos en Cassette o Diskette solicitándolos directamente a FERRE MORET, S.A., o en sus distribuidores

Este libro ha sido traducido por Dn. Luis Alvarez Muñoz, estudiante de ingeniería de telecomunicación y experto conocedor del ZX Spectrum.

Imprime: APSSA, ROCA UMBERT, 26 - L'HOSPITALET DE LL. (Barcelona)

Depósito legal B-42.504/85

ISBN 84-86437-19-9

Advertencia importante

Los circuitos, procedimientos y programas reproducidos en este libro, son divulgados sin tener en cuenta el estado de las patentes. Están destinados exclusivamente al uso amateur o docente, y no pueden ser utilizados para fines comerciales.

Todos los circuitos, datos técnicos y programas de este libro, han sido elaborados o recopilados con el mayor cuidado por el autor y reproducidos utilizando medidas de control eficaces. No obstante, es posible que exista algún error. FERRE MORET,S.A. se ve por tanto obligada a advertirles, que no puede asumir ninguna garantía, ni responsabilidad jurídica, ni cualquier otra responsabilidad sobre las consecuencias atribuibles a datos erróneos. El autor les agradecerá en todo momento la comunicación de posibles fallos.

ESTRUCTURACION

==============

Página

Página

Pagina

Prefacio



Los ordenadores como el SPECTRUM no sólo están para jugar. Por muy fascinantes que puedan ser los programas elaborados -juegos de fútbol, guerras espaciales, carreras de coches, partidas de ajedrez, luchas contra extraños monstruos, etc.- no hay nada más excitante que ser uno mismo el que programa. Quien analiza por primera vez un problema y consigue que el programa introducido lo resuelva en cuestión de segundos y sin errores, puede congratularse. Un logro de este tipo prueba la utilidad del ordenador, es decir, que nos libere del pesado trabajo mental, siendo ésta su principal aplicación, que determina su creciente importancia.

Por este motivo, en el presente libro queremos enseñarle al alumno cómo aplicar el ordenador, en concreto el SPECTRUM, para resolver los problemas fundamentales de la enseñanza escolar.
En resumidas cuentas, con este libro se persiguen simultáneamente cuatro objetivos:

1. Se toman los problemas escolares más normales, por ejemplo de matemáticas, física u otras asignaturas y se proponen medios para una solución.

2. La descripción de estos medios es presentada al lector de forma que pueda aplicarlos al ordenador.

3. Se informará al lector sobre la utilización del lenguaje de programación BASIC destinado a aquellos programas necesarios para solucionar los problemas antes mencionados.

4. De esta forma, el lector dispondrá de una colección de programas, que le será útil en la escuela.

Comprenderá el lector que la utilidad de este libro se hace patente especialmente cuando prueba él mismo los programas presentados al igual que cuando ensaya con modificaciones de éstos o con otros programas.

Como en muchos otros ámbitos, dentro del campo de la programación y de la utilización de ordenadores se cumple lo siguiente:

'' La práctica hace al maestro ''

En los capítulos de este libro, se cubren los siguientes campos:

1. Problemas de matemáticas
2. Problemas de química
3. Problemas de física
4. Ejercicios de lenguas
5. Problemas de biología/ecología
6. Problemas de geografía e historia
7. Problemas de economía
8. Otros problemas

En estos capítulos se contienen algunas explicaciones acerca de las bases y condiciones de aplicación del ordenador y se discuten, también, los elementos básicos del lenguaje de programación BASIC.
Los lectores que posean suficientes conocimientos de BASIC pueden prescindir de estas explicaciones.
Debemos hacer la siguiente aclaración con respecto a los programas que desarrollamos y presentamos en los capítulos siguientes:

Por una parte, se trata de programas que deben resolver determinados problemas (Ejemplo: Programa para calcular un '' máximo común divisor '').

Estos programas pueden " alimentarse " con diferentes datos de partida, y sirven siempre para el mismo fin. Por otro lado, existen también programas sólo de entrenamiento: Por ejemplo, un programa que sirve para aprender vocablos ingleses, que no resuelve ningún problema, sino que sólo "entrena".
En todo caso, estos programas tienen aplicaciones más generales cuando se proveen de material de partida más variado.

Se está preparando una cassette de programas para este libro, donde se encuentran todos los programas aquí discutidos, de modo que el lector que quiera utilizarlos se pueda ahorrar el trabajo de teclearlos.
La siguiente observación es de gran importancia para las próximas explicaciones:

En los programas donde se resuelven los problemas del colegio hemos prescindido conscientemente de utilizar los elementos más avanzados y complejos del lenguaje BASIC del SPECTRUM. Por el contrario, nos hemos esforzado en utilizar los elementos básicos y sencillos del BASIC. El motivo es evitar sobrepasar, tanto las posibilidades del lector como las del principiante, en materia de programación y conseguir, de este modo, que tenga la ocasión de centrarse en los problemas.

Esta limitación tiene como consecuencia el que algunos programas parezcan demasiado simples pero, aun con los medios más sencillos funcionan y éste era nuestro objetivo.

No obstante, los programas más complejos de los últimos capítulos, requieren la aplicación de otros elementos sintácticos, que serán tratados en caso concreto.

Rogamos también al lector tenga en cuenta que algunos de los programas presentados figuran simplemente a modo de ejemplos:

Si, por ejemplo, en un programa de repaso de vocablos sólo se prevén 10, el lector, para poder utilizarlo de forma concreta, deberá ampliar primero la lista de éstos. Este procedimiento naturalmente no modifica la estructura básica del programa -que es lo que a fin de cuentas importa-.



Capítulo 1 : Elementos básicos del lenguaje BASIC

1.1 Consideración previa

En este capítulo, se presentan y se explican además de algunos conceptos básicos del proceso de datos, que volverán a aparecer más tarde, los elementos básicos más importantes del lenguaje BASIC.
Aquellos lectores que ya conozcan suficientemente el BASIC pueden saltarse este capítulo.

Sin embargo, los lectores que no tengan ninguna experiencia como programadores pensarán que las explicaciones de este capítulo son algo someras. A éstos les diremos que en los próximos capítulos se aclarará la utilización, el funcionamiento y el efecto de estos elementos sintácticos mientras se aplican de forma práctica.
Visto de este modo, este capítulo contiene sólo una introducción.

1.2 Conceptos básicos que deberían conocerse

Un ordenador moderno no sólo sabe calcular. Sabe, asimismo, manipular letras, palabras y textos y puede, entre otras cosas, generar representaciones gráficas,producir música, así como ejercer tareas de control. Ello significa que los ordenadores, que hoy en día ya pueden adquirirse por un precio bastante módico, no sólo disponen de la misma capacidad de cálculo que un ordenador grande de los años sesenta, que era económicamente inaccesible, sino que también su aplicación es más universal.

Datos : Bajo "datos" puede entenderse en el sentido más amplio la palabra "informaciones". Concretamente puede tratarse de:

- cifras y números
- valores
- letras
- símbolos (caracteres especiales)
- palabras y textos

Proceso : Bajo "proceso de datos" se entienden los procedimientos (frecuentemente cálculos), que sirven para registrar estos datos, para almacenarlos, para valorarlos o analizarlos y para ofrecer resultados.

Cuando el proceso de datos en la forma descrita no se realiza manualmente, sino con la ayuda de ordenadores, es necesario conocer otros conceptos nuevos:

bit : Bajo bit "binary information digit" se entiende la unidad de información más pequeña (ver antes : "datos" = "informaciones").
La unidad de información más pequeña se llama bit, porque un bit puede tener únicamente dos contenidos distintos. Visto desde la perspectiva del fabricante de ordenadores esto significa: un bit puede almacenar sólo dos contenidos de información. Estos contenidos se denominan usualmente 0 y 1.

byte : El byte es la reunión de una serie de bits. Son ocho los bits que constituyen un byte. De este modo, un byte está en condiciones de almacenar una serie de 8 ceros y/o unos.

Carácter : Distinguimos tres grupos de caracteres
1. caracteres numéricos = cifras
2. caracteres alfabéticos = letras
3. otros caracteres = caracteres especiales

Generalmente, cada uno de estos caracteres se representa en una serie de 8 ceros y/o unos en el ordenador, es decir, que un byte acepta un carácter.

Campo : Una serie de caracteres (p.e. palabras o números) constituyen un campo.

La palabra PRINT, por ejemplo, consta de 5 caracteres, ocupa 5 bytes en el ordenador, y como estos 5 símbolos pertenecen a una misma palabra, constituyen un campo.

Lo mismo ocurre con el número 178, que es almacenado en un campo de tres posiciones (se deduce que también pueden existir campos de una sóla posición).

Variable: Una variable es una magnitud que puede tomar diversos valores - precisando - un nombre, bajo el cual pueden seleccionarse diversos valores (ver abajo).
Por ejemplo, bajo el nombre ''tamaño'' podemos hallar los valores 178, 185, 167 etc., detrás del nombre ''número de niños'' los valores 1, 2, 0, 4 etc., bajo el nombre ''localidad'', por ejemplo los valores ''Palencia'', ''Alcántara'', ''Palafrugell''.

Valor : Como puede deducirse de la definición anterior, bajo "valor" entendemos una determinada característica que puede adquirir una variable dada. No es necesario que sean números, también puede tratarse de palabras o textos como en el ejemplo de la variable anterior "localidad".

Cada uno de estos valores ocupa un campo del ordenador (ver arriba). También puede llamarse posición de memoria.

String : Llamamos "cadena" a un texto que puede representar un valor de una variable (ver arriba); genéricamente se trata de una cadena de caracteres.

Tabla : La totalidad de los valores que puede tomar una variable (números o palabras) se llama "tabla".

Registro: Varios campos relacionados constituyen un registro. Por ejemplo, todos los datos referentes a una persona (nombre, sexo, edad, domicilio, etc.) forman un registro. En este ejemplo, el registro es una serie de números y/o palabras relacionados entre sí.
Una serie de palabras solas puede también constituir un registro, por ejemplo la instrucción:

```
PRINT   "PEPE", "ANA", "MAXIMO"
```

El registro se denomina también "record".

Fichero: Varios registros relacionados forman un fichero. Obtendremos un fichero por ejemplo si confeccionamos los registros de todos los empleados de una empresa. Podemos obtener igualmente un fichero si contemplamos todas las instrucciones de un programa en conjunto
El fichero se llama también "file".

1.3 Referente al funcionamiento de los ordenadores

Ya explicamos lo que se entiende bajo "proceso de datos". De ello se deduce qué actividades (funciones) debe realizar un equipo de proceso de datos:

1. Recepción de datos (entrada de datos)		
2. Almacenamiento de datos	E	entrada
3. Proceso de datos propiamente dicho (p.e. cálculos o traslado de textos y parecidos)	P	proceso
4. Salida de datos	S	salida

La entrada de información, su procesado y su salida hacen necesarios una serie de pasos organizativos , que no es necesario tratar aquí, porque generalmente los resuelve el mismo ordenador. De ello se encarga el llamado sistema operativo.
Así, el ordenador debe obtener nuestras informaciones, para que el proceso de los datos pueda ponerse en marcha y proporcionar resultados.

Las informaciones que debemos darle se clasifican en tres grupos:

1. Los datos a procesar
2. Las instrucciones
   (Pasos, que son necesarios para someter los datos introducidos a los procedimientos de valoración requeridos)
3. Comandos
   (Informaciones e instrucciones que damos al sistema operativo del ordenador; ver arriba)

De acuerdo con los diferentes grupos de cometidos que debe resolver el ordenador distinguiremos entre:

1. Aparatos de entrada (p.e. el teclado)
2. Ambito del procesado (la llamada unidad central)
3. Aparatos de salida (pantalla o impresora)

Aparte de esto existen también las llamadas memorias externas (o periféricas). Sirven para almacenar a largo plazo los programas y los datos. Generalmente se trata de la cassette y de los llamados diskettes, que se reproducen o leen en los correspondientes aparatos reproductores.

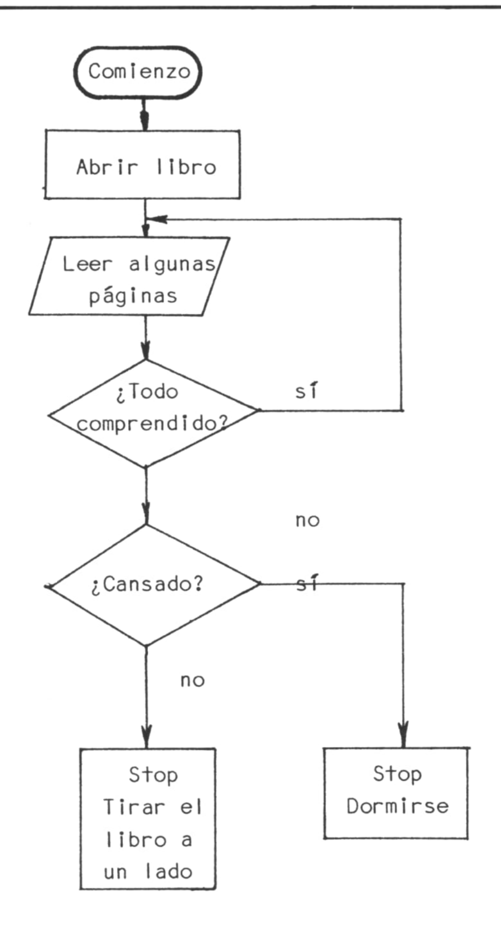
1.4 Análisis de problemas

Resolver un problema concreto con ayuda de un ordenador sólo es posible si se ha hallado de antemano su solución. El lector deberá recordar siempre esta evidencia. Esto quiere decir que la solución, a pesar de la modernidad de los ordenadores, sólo puede hallarse mentalmente y no en la máquina; el ordenador únicamente se ocupa de realizar los pasos necesarios.

Para que un ordenador pueda efectuar esta tarea debe indicársele el procedimiento de resolución en forma de instrucciones individuales. Esto lleva consigo que deba descomponerse mentalmente el problema a resolver en una sucesión lógica de pasos individuales - este procedimiento se denomina análisis del problema.

En un análisis de problema resulta útil representar la sucesión de pasos gráficamente en forma de un diagrama de flujo.

Veamos el siguiente ejemplo ilustrativo:



Fuente : W. VOSS : BASIC aprendido rápida- y fácilmente con el Alphatronic PC, Darmstadt 1984, (Heim-verlag), pag. 8.

En los capítulos siguientes utilizaremos una y otra vez el instrumento ''análisis del problema'', porque de este modo, la programación, es decir, la redacción de las secuencias de instrucciones para el ordenador, se facilita considerablemente, evitándose los errores de programación.

1.5 Visualización de resultados

Se le denomina programa a una sucesión de instrucciones que hacen que un ordenador resuelva por sí mismo un problema dado. Debe ser escrito en un lenguaje que el ordenador pueda comprender. Este es el BASIC.

```
                Beginner's
                All-Purpose
BASIC    =      Symbolic
                Instruction
                Code
```

Todo programa BASIC consta de una sucesión de instrucciones (statements) que deben ser introducidos línea a línea en el ordenador. Una línea se llama sentencia.

Regla 1: Cada sentencia BASIC requiere un número.

Regla 2: Cada sentencia debe concluirse pulsando la tecla ENTER.

Regla 3: Una sentencia puede contener varias instrucciones, que hay que separar mediante ":"

Instrucción 1:

```
nn   STOP
```

La instrucción STOP cierra el programa BASIC. Sirve asimismo para separar una subrutina del resto del programa BASIC.

Instrucción 2:

```
               ⎧ Sin argumento
               ⎪ Número
nn  PRINT     ⎨ Operación
               ⎪ Cadena
               ⎩ Nombre de variable
```

La instrucción PRINT trae los resultados a la pantalla. Se distinguen cinco posibilidades:

1. Sin argumento : Se obtiene una línea en blanco
2. Número : Se visualiza el número
3. Operación : Se visualiza el resultado de una operación
4. Cadena : Se visualiza la cadena (cadena de caracteres puesta entre comillas)
5. Nombre de variable : El contenido de este campo de la memoria es visualizado

Puede utilizarse el signo ' (apóstrofe), para dejar líneas en blanco

El siguiente programa BASIC utiliza todas estas posibilidades:

```
10 LET x=88888888
20 PRINT 3
30 PRINT
40 PRINT 4.8/2
50 PRINT
60 PRINT "Susi"
70 PRINT
80 PRINT x
```

Este programa BASIC sólo se ejecuta si tecleamos el comando RUN:

Comando 1

RUN

Para el ejemplo anterior, después del comando RUN obtendremos en pantalla los siguientes resultados:

```
3

2.4

SUSI

88888888
```

Regla 4: Los comandos no reciben ningún número de sentencia

Si después de la impresión del resultado deseamos ver nuevamente nuestro programa en pantalla, necesitamos el siguiente comando:

Comando 2

LIST

La instrucción PRINT requiere algunas puntualizaciones adicionales:

Las diferentes posibilidades enunciadas en la instrucción 2 pueden utilizarse tanto mezcladas entre si como en una sola instrucción PRINT.
En este caso deben separarse mediante comas y puntos .

Regla 5: Si se separan con comas las diferentes expresiones dentro de la instrucción PRINT, la impresión prosigue por el principio de la siguiente zona de la pantalla (columna 16) ; cuando se utiliza el punto y coma, la impresión se produce justo a continuación.

Regla 6: Por regla general, toda instrucción PRINT produce un salto de una línea. Si la instrucción contiene la palabra PRINT sola, se produce un único salto de línea.

Regla 7: Una coma o un punto y coma al final de una instrucción PRINT reprimen el salto de línea.

La instrucción PRINT genera impresiones de resultados, que comienzan en el margen izquierdo de la pantalla. Para cambiar esto pueden utilizarse las sentencias auxiliares AT y TAB:

Instrucción 3:
----

nn PRINT AT S,Z;valor al imprimir

Esta instrucción visualiza a partir del número de línea S y de columna Z de la pantalla especificado. S pueden tomar valores entre 0 y 21, Z puede tomar valores entre 0 y 31. Para ello debe imaginarse la pantalla dividida en treinta y una columnas y 21 líneas, situándose la línea 0 arriba y la columna 0 a la izquierda.
Utilizando los comandos PRINT ≠0, y PRINT ≠1, puede imprimirse en las líneas 22 y 23 de la pantalla, (estas líneas las reserva el ordenador para imprimir informes)

Instrucción 4:
----

nn PRINT TAB X;valor a imprimir

Esta sentencia auxiliar se utiliza conjuntamente con PRINT y tabula la impresion en el formato deseado. X puede tomar valores comprendidos entre 0 y 65535, aunque la salida en pantalla se producirá entre las columnas 0 y 31.

Para borrar un programa que ya no nos es de utilidad antes de introducir uno nuevo, se requiere el comando siguiente, que borra el contenido de memoria del program y pone a cero todas las variables: NEW

En el siguiente punto trataremos el concepto del nombre de variable, que ya jugaba un papel en la instrucción PRINT, además hablaremos sobre particularidades de las operaciones realizables.

1.6 Asignaciones de valores

Como ya tendremos ocasión de ver, resulta muy útil para su aplicación, el dotar de nombres a las posiciones de memoria, con el fin de almacenar diversos valores bajo estos nombres.

En principio basta con distinguir entre dos tipos de variable:

1. Variables numéricas
2. Variables de cadena

Las variables numéricas pueden tomar números como valores, como por ejemplo:

3.5   4   -17.01   7   18.   -3   0.04   0

-7,5   20   etc.

Las variables de cadena toman como valor secuencias de caracteres, que deben colocarse entre comillas, por ejemplo:

"ANTONIO"   "00"   "SEVILLA"   "B-30"   etc.

Regla 8: Los nombres de las variables numéricas pueden estar constituidos por distinta cantidad de caracteres, el primero de los cuales debe ser forzosamente una letra. No existe ninguna diferencia entre dos variables numéricas escritas en letras mayúsculas o minúsculas. Los nombres de las variables de cadena deben ser forzosamente de un sola letra, seguida del símbolo $

Ejemplos:
----

1. Variables numéricas : X B1 Peso
2. Variables de cadena : F$ C$ Z$

Después de aclarar estos conceptos hablaremos sobre la instrucción LET, que sirve para asignar valores de modo que pueda disponerse de los mismos a lo largo de todo el programa:

Instrucción 5:

```
                                    número
nn   LET   nom. variable   =  {     operación
                                    cadena
                                    nombre de variable
```

Esta instrucción permite calcular de la misma manera que se hacía con la instrucción PRINT.

Los operadores para calcular son los siguientes:

+	Adición
-	Sustracción
*	Multiplicación
/	División
-	Cambio de signo
↑	Potenciación

La prioridad disminuye en sentido descendente.

El rango de los operadores es análogo al que conocemos de la escuela; en caso necesario deben colocarse paréntesis.

Así, la instrucción LET nos permite escribir sentencias más complicadas. En caso de error a la hora de teclear, la forma más sencilla para corregir, es cerrar la sentencia errónea con ENTER y escribir simplemente la frase de nuevo. Para borrar del todo una sentencia, hay que teclear su número y pulsar ENTER.

Para insertar a posteriori una sentencia en un programa, es necesario introducir ''números de sentencia intermedios''.

Muchos cálculos pueden simplificarse mediante las llamadas funciones incorporadas. Las más importantes son:

Funciones importantes

Nombre	Cometido
SIN (X)	Calcula el seno de X en radianes
COS (X)	Calcula el coseno de X en radianes
TAN (X)	Calcula la tangente de X en radianes
LN (X)	Calcula el logaritmo neperiano de X
EXP (X)	Ofrece la funcion exponencial en base e
SQR (X)	Ofrece la raiz cuadrada de X
ABS (X)	Ofrece el valor absoluto de X
SGN (X)	Ofrece 1, -1 o 0 segun el signo de X
INT (X)	Ofrece la parte entera de X
RND (X)	Generación de números aleatorios
CHR$ (X)	Ofrece el caracter cuyo codigo ASCII es X
LEN (X$)	Ofrece el número de caracteres de X$
ATN (X)	Calcula la arcotangente de X
ASN (X)	Calcula el arcoseno de X
ACS (X)	Calcula el arcocoseno de X
PI	3.141592654

1.7 Entrada de informaciones

Para introducir informaciones resulta muy adecuada la siguiente instrucción:

Instrucción 6:

```
nn   INPUT   Lista de var.
```

Esta instrucción puede parecer complicada, pero es fácil de comprender:

Cuando el ordenador llega a esta instrucción, interrumpe el proceso del programa y espera a que el usuario introduzca tantos valores, (separados por comas), como nombres haya en la lista de variables (es decir, como mínimo un valor, en el caso de que se trate de un solo nombre).

Ejemplos:

INPUT X	El ordenador pide un número, que será almacenado en el campo X.
INPUT A,B,C	El ordenador pide tres números.

INPUT N$ El ordenador pide una cadena (p.e. "ANTONIO"), que será almacenada en el campo N$.

Regla 9: Los nombres en la lista de variables de la instrucción INPUT se separan mediante comas Los valores a introducir también hay que separarlos de la misma forma.

Es posible presentar en pantalla un mensaje informativo que indique al usuario que tipo de dato debe introducirse. Esto se logra escribiendo el INPUT del siguiente modo:

INPUT "Comentario........";A

Ejemplo:

```
10 PRINT "CUATRO VALORES, POR FAVOR"
20 INPUT A,B,C,D
30 LET S=A+B+C+D
40 LET AM=S/4
50 PRINT
60 PRINT "PROMEDIO = ";AM
```

Vemos claramente como este programa calcula el promedio de cuatro valores a introducir.

1.8 Bifurcaciones del programa

Mientras en un programa BASIC no aparezca ninguna bifurcación (llamada salto), el proceso tiene lugar en el orden de los números de sentencia asignados. Si no se desea respetar este orden son necesarios los saltos en el programa.
Distinguimos entre saltos condicionales y saltos no condicionales.

Veámos en primer lugar el salto condicional, que se realiza gracias a la siguiente instrucción:

Instrucción 7:



Esta instrucción obliga al sistema operativo del ordenador a consultar: Si la condición expresada después del término IF se cumple, entonces se ejecuta la instrucción escrita detrás de THEN o se pasa a la sentencia número mm.

De no cumplirse esta condición, el control del programa pasa a la sentencia que sigue a la sentencia IF.

Regla 10: Si se cumple la sentencia IF, se efectúa la instrucción detrás del THEN o se salta al número de sentencia ahí escrito.
De no cumplirse esta condición, se procesa la siguiente sentencia.

Regla 11: Si hay otras instrucciones detrás del IF dentro de la misma sentencia, sólo se se realizarán si se cumple la condición.

Ejemplo:

```
10 LET I=1
20 LET Q=I*I
30 LET W=SQR (I)
40 PRINT AT I,0;I;AT I,11;Q;AT I,22;W
50 LET I=I+1
60 PRINT
70 IF I<=20 THEN  GO TO 20
80 PRINT #0;"       FIN DEL PROGRAMA"
90 PAUSE 100
```

Este programa calcula los cuadrados (que aparecen en el campo Q; línea 20) de todos los números enteros del 1 al 20 (que aparecen en el campo I), y también sus raíces cuadradas (en el campo W; línea 30), imprimiendo cada grupo de tres valores en una línea de la pantalla.

La línea 70 resulta decisiva. Dicta que el programa vuelva una y otra vez a la línea 20, mientras en el campo I siga habiendo un valor menor de 21 tras elevar su contenido cada vez en 1 (línea 50).

En tanto esta condición se cumpla, se ejecutará la instrucción que sigue al THEN (que es el salto de vuelta a la línea 20); cuando ya no se cumpla se procesará la instrucción que sigue al IF (línea 80).

La alternativa al salto condicional la encarna el salto no condicional:

Instrucción 8:

```
nn   GOTO   mm
```

Cuando el control del programa alcanza esta instrucción tiene lugar un salto a la línea número mm.
Para el siguiente programa de muestra, necesitamos otra instrucción:

Instrucción 9:

```
DIM   nombre de var. (número)
```

Ejemplo:

```
DIM   X (20)
```

En este caso la matriz X (unidimensional), se dimensiona con un total de 20 elementos. El subíndice de identificación de cada elemento variará en este caso entre 1 y 20.

Para dimensionar matrices de cadena debe mantenerse libre un campo por cada carácter de la cadena, por ejemplo:

DIM A$ (5)

Mantiene libres cinco campos para una cadena, de forma que pueda introducirse una cadena con una longitud máxima de cinco caracteres.

```
  10 DIM A(20)
  20 FOR N=1 TO 20
  30 PRINT "DEME EL VALOR DE ";"A";(N)
  40 INPUT A(N)
  50 CLS
  60 NEXT N
  70 PRINT  FLASH 1; INK 0; PAPER 4;"    ELEMENTOS DE LA MAT
RIZ A    "
  80 PRINT : PRINT : PRINT : PRINT
  90 FOR N=1 TO 20
 100 PRINT "A";"(";N;")";" = ";A(N),
 110 NEXT N
 120 PRINT #0;"       FIN DEL PROGRAMA"
 130 PAUSE 200
```

Este programa es un ejemplo de dimensionado de una matriz numérica. Se parte de una instrucción de dimensionado en la línea 10. A continuación, el programa pide que asigne un valor a cada elemento de la matriz mediante un bucle, (líneas 20 a 60). Una vez se han pedido los valores de los 20 elementos, el programa visualiza uno por uno todos los elementos de la matriz unidimensional A. Esto lo efectúa el bucle situado entre las líneas 90 y 110.

1.9 Bucles

A menudo resulta adecuado ejecutar determinadas partes del programa más de una vez. Para ello se necesitan los llamados bucles (loops). Estos bucles pueden generarse sencillamente con las instrucciones tratadas hasta el momento, tal como puede apreciarse en el programa anterior.

Las siguientes instrucciones simplifican la constitución de bucles, abriéndose las siguientes posibilidades adicionales.

Instrucción 10:

```
nn  FOR  variable index = valor inicial
    TO   valor final
    STEP incremento
```

La siguiente instrucción va unida a la anterior:

Instrucción 11:

```
nn  NEXT nombre variable indexada
```

El valor inicial, el valor final y la longitud pueden ser números, variables o expresiones aritméticas.

''Cuando el ordenador llega a la instrucción FOR, se iguala la variable indexada al valor inicial y luego se ejecuta esta parte del programa situada entre FOR y NEXT con este valor.
Tras alcanzar el NEXT, la variable indexada salta al siguiente valor, que resulta del valor inicial + la longitud y se ejecuta nuevamente la misma parte del programa, esta vez con el segundo valor de la variable indexada.
Esto sigue hasta que, con una nueva adición de la longitud tras alcanzar el NEXT, se supera el valor final. Entonces se ejecuta la línea que sigue al NEXT.
Puede suprimirse el "STEP longitud" cuando la longitud deba ser igual a uno.''

(PRUST, 1982, pag. 64).

El programa siguiente imprime 10 veces la expresión "BUENOS DIAS" de forma uniforme, puesta una debajo de la otra:

```
10 FOR I=1 TO 10
20 PRINT "          BUENOS DIAS"
30 PRINT
40 NEXT I
50 PRINT #0;"       FINAL DEL PROGRAMA"
60 PAUSE 100
```

1.10 La utilización de memorias externas

Siempre se desea guardar por un espacio de tiempo cualquier programa elaborado, de modo que se pueda utilizar en caso de necesidad (del mismo modo, en ocasiones se desean utilizar programas de otros autores). Esto es posible utilizando cintas de cassette o diskettes. Para ello deben conocerse los siguientes comandos.

Para cargar un programa BASIC a partir de un cassette necesitamos el :

Comando 3:

```
LOAD "Nombre del programa"
```

Para grabar un programa BASIC en un cassette necesitamos el:

Comando 4:

```
SAVE "Nombre del programa"
```

El cassette no es un recurso muy fiable. Para verificar si la grabación ha resultado correcta se usa el comando siguiente:

```
VERIFY "Nombre del programa"
```

Se puede forzar la autoejecución de un programa, inmediatamente después de su carga, y sin necesidad de pulsar RUN, mediante la siguiente instrucción:

```
SAVE "Nombre del programa" LINE X
```

Donde X es la línea del programa desde donde se desea que se inicie el programa.

Si al escribir el comando LOAD no especificamos el nombre del programa, es decir, si escribimos LOAD "", el ordenador cargará el primer programa que se encuentre. Si por el contrario, especificamos el nombre del programa el ordenador solo cargará el programa cuyo título se corresponda con el que hemos escrito entre las comillas, e ignorará el resto de los programas que pudiera contener el cassette - aunque visualizará el nombre de los mismos a medida que se sucedan. En ningun caso el nombre del programa podrá superar la cantidad de diez caracteres.
Existen también otros comandos que permiten la carga o la grabación de información con distintos formatos del habitual.

Estos comandos permiten la carga, grabación y verificación de programas en código máquina.

```
  LOAD "Nombre del programa" CODE dirección,longitud
  SAVE "Nombre del programa" CODE dirección,longitud
VERIFY "Nombre del programa" CODE dirección,longitud
```

Estos comandos permiten la carga o grabación de la pantalla de presentación.

```
LOAD "Nombre de la pantalla" SCREEN$
SAVE "Nombre de la pantalla" SCREEN$
```

Los siguientes, permiten la carga o grabación de UDGs:

```
LOAD "Nombre UDG" CODE USR "A-Z"
SAVE "Nombre UDG" CODE USR "A-Z"
```

Para la grabación y carga de matrices se utilizan los comandos:

```
LOAD "Nombre" DATA "letra" ( )
SAVE "Nombre" DATA "letra" ( )
```

Finalmente la sentencia MERGE permite mezclar programas BASIC con distinta numeración.

1.11 Otros aspectos

Para cerrar este capítulo introductorio veámos unos pocos y útiles aspectos aadicionales.
La presentación de un programa mejora sustancialmente si durante la ejecución del mismo y antes de la impresión de los resultados, se borra la pantalla. Esto es posible gracias a la instrucción:

```
nn  CLS
```

A menudo resulta necesario parar el proceso de un programa en casos como la programación del llamado bucle sinfin. Pulsando las teclas CAPS SHIF + BREAK SPACE se para la ejecución del programa.

Finalmente quiero mencionaremos la instrucción

Instrucción 12:

```
nn   STOP
```

mediante la cual el desarrollo del programa se interrumpe por si mismo. Para activar nuevamente el programa interrumpido necesitamos el comando:

Comando 5:

```
CONT
```

Finalmente quisiera añadir que a menudo resulta útil incluir observaciones explicativas en un programa (títulos, comentarios, explicaciones, etc.). Esto es posible con la siguiente instrucción:

Instrucción 13:
----

nn REM texto

Esta instrucción no influye en la ejecución del programa, pero aparecerá en cualquier listado que se efectúe del mismo.

Espero que estas indicaciones sirvan para que incluso el principiante tenga una idea de los elementos básicos del lenguaje BASIC. De esta manera estará en condiciones de comprender las aplicaciones que siguen. Si aún así tuviera problemas de comprensión, sólo puedo darle un consejo:

P R O B A R

Probando los programas e intentando analizar por qué el ordenador hace precisamente una cosa y no otra se aprende más que de cualquier otra forma.



Capítulo 2 : Matemáticas

2.1 Consideración previa

Si queremos una buena aplicación para el ordenador, no hay nada más indicado que las matemáticas. Por esta razón empezaremos por ellas. Más tarde veremos, no obstante, que las matemáticas no son el único campo de aplicación.

En todo caso, buscaremos unos ejemplos adecuados pues todos sabemos que las matemáticas cubren un espectro de grados de dificultad muy amplio - empezando con las operaciones básicas y pasando por el cálculo infinitesimal.

Tratarmos problemas de "dificultad media", es decir, nos orientaremos hacia problemas como los que se presentan en el último nivel de EGB y en el primer curso de bachillerato.

Las matemáticas son perfectas para demostrar las posibilidades del ordenador, sin que la presentación de los problemas tratados ocupe demasiado tiempo debido a un eventual grado de complejidad de los mismos.

Este libro no pretende exceder en las posibilidades del lector con los ejemplos de aplicación, simplemente se trata de presentar la capacidad del ordenador. Esto se conseguirá con problemas de estructura sencilla, adaptados a los problemas de los escolares más jóvenes.

En los programas de este y de los siguientes capítulos se seguirá la siguiente norma :

> ¡En los programas prescindiremos de corregir
> los errores de introducción del usuario!

Por ejemplo, en un problema matemático dado, si sólo pueden utilizarse números positivos (por ejemplo en las raíces), el correspondiente programa no tiene previsto el informarle al usuario en caso de haber introducido por error un número negativo.

Comenzaremos ahora con un ejemplo de grado de dificultad "medio".

2.2 Teorema de Pitágoras

Por medio del famoso teorema de Pitágoras puede mostrarse de forma clara, cómo hay que estructurar un problema.

Siempre se seguirán los siguientes pasos de trabajo:

Pasos de trabajo:

1. PASO : Presentación del problema

2. PASO : Análisis del problema

3. PASO : Diagrama de flujo

4. PASO : Programa

5. PASO : Lista de variables

6. PASO : Descripción del programa

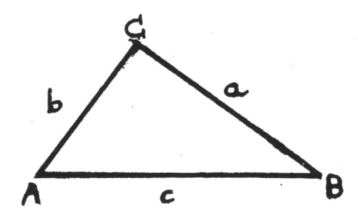
7. PASO : Resultados

Siguiendo esta estructuración pasemos ahora al problema citado:

1. PASO : Presentación del problema

El teorema de Pitágoras dice que en el triángulo rectángulo, la longitud del lado base (hipotenusa) es igual a la raiz cuadrada de la suma de los cuadrados de ambos catetos.

Esta figura ilustra lo dicho:



$$c = \sqrt{a^2 + b^2}$$

2. PASO : Análisis del problema

En este caso, el análisis del problema es muy sencillo:

Debe recordarse tanto aquí como en los siguientes problemas que cualquier proceso de datos funciona según el

Principio E P S

ya presentado.

E	=	Entrada
P	=	Proceso
S	=	Salida

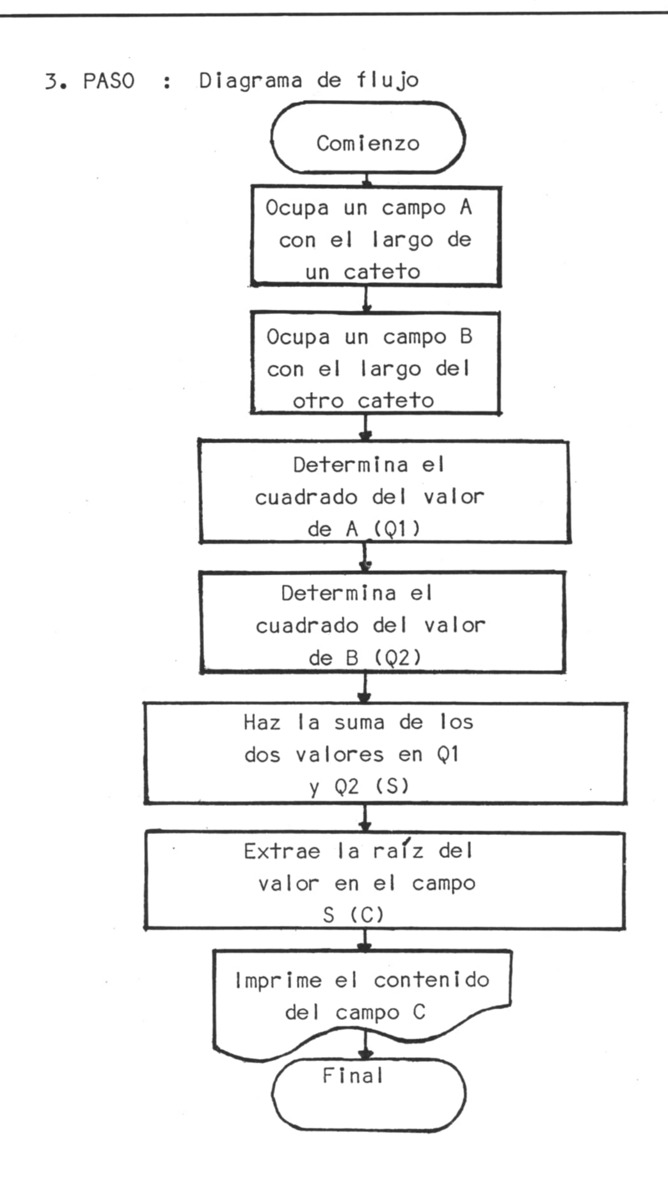
Por esta razón deberemos fijar, en primer lugar, las informaciones que debe recibir el ordenador de nuestra parte, para la resolución del problema.

Las informaciones de entrada son lógicamente las longitudes de ambos catetos a y b.

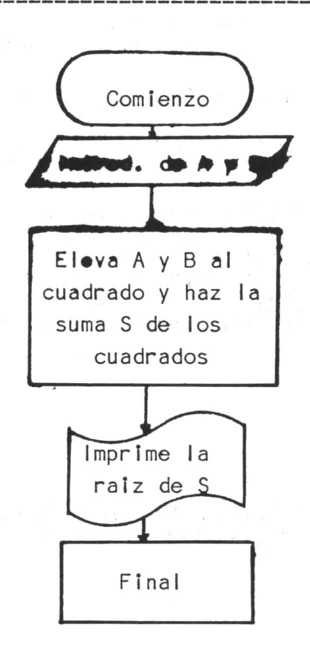
En el proceso, estos dos valores deben ser elevados al cuadrado; debe efectuarse su suma y finalmente extraerse la raiz cuadrada de esta suma.

La salida consiste únicamente en visualizar en la pantalla el resultado obtenido (la longitud de la hipotenusa c).

3. PASO : Diagrama de flujo



Versión resumida del diagrama de flujo:



```
  10 REM     PITAGORAS
  20 BORDER 0: PAPER 0: INK 7: CLS
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA DETERMIN
AR LA HI-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"POTENUSA DE UN TRIANGUL
O RECTAN-"
  50 PRINT  PAPER 4; INK 0; FLASH 1;"GULO.
       "
  60 PRINT : PRINT "    -----------------------  "
  70 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  80 PRINT "    -----------------------  "
  90 INPUT "PRIMER CATETO = ?  ";A
 100 PRINT AT 9,5;"PRIMER CATETO   =  ";A
 110 INPUT "SEGUNDO CATETO = ?  ";B
 120 PRINT AT 11,5;"SEGUNDO CATETO  =  ";B
 130 LET Q1=A*A
 140 LET Q2=B*B
 150 LET S=Q1+Q2
 160 LET C=SQR (S)
 170 FOR N=0 TO 10
 180 BEEP .06,N
 185 NEXT N
 190 PRINT AT 16,3; PAPER 1; INK 6;"HIPOTENUSA   =  ";C
 200 PAUSE 40: PRINT #0; FLASH 1; PAPER 6; INK 4;"        FI
N DEL PROGRAMA        "
 210 PAUSE 200
```

5. PASO: lista de variables

A	:	Primer cateto
B	:	Segundo cateto
C	:	Hipotenusa
Q1	:	Cuadrado del primer cateto A
Q2	:	Cuadrado del segundo cateto B
S	:	Suma de los cuadrados de los catetos

6. PASO : Descripción del programa

Nos basamos en la primera versión, la más larga del programa:

Línea 10 : Comentario

Línea 20 : Se establecen los colores de la pantalla y se procede a su borrado.

Línea 30-80: Impresión del título en pantalla

Línea 90 : Se pide un valor para el primer cateto, que es almacenado en el campo A

Línea 110 : Lo mismo con el segundo cateto en el campo B

Línea 130 : Cálculo del cuadrado del primer cateto

Línea 140 : Cálculo del cuadrado del segundo cateto

Línea 150 : Suma de los dos cuadrados resultantes

Línea 160 : Cálculo de la raiz de la suma

Línea 170-185: Bucle sonoro

Línea 190 : Impresión del resultado en pantalla

Línea 200 : Final del programa

7. PASO : Resultados

Una vez iniciado el programa (en su primera versión) aparece el título en la pantalla:

```
PROGRAMA PARA DETERMINAR LA HIPOTE-
  NUSA DE UN TRIANGULO RECTANGULO
      ----------------------
      PROF. DR. W.VOSS, 1984
      ----------------------
```

Más abajo se pide el primer valor:

```
PRIMER CATETO  :  ?
```

A continuación, introduzcamos por ejemplo el valor 3 y el ordenador reaccionará de la forma siguiente:

```
SEGUNDO CATETO  :  ?
```

Introduzcamos por ejemplo el valor 4, a lo que el ordenador contestará:

```
HIPOTENUSA C  =  5
```

Este programa es adecuado para calcular, a partir de cualquier valor, la hipotenusa de un triángulo rectángulo. En este sentido se trata de un programa muy general.

Hemos tratado este problema muy extensamente con el fin de ilustrar el procedimiento a seguir en cada uno de los ejemplos. De ahora en adelante nos ahorraremos explicaciones, por lo menos cuando se trate de ejemplos tan sencillos como el Teorema de Pitágoras.

2.3 M.C.D. y m.c.m.

1. PASO : Presentación del problema

En muchos cálculos algebraicos resulta útil conocer el máximo común divisor (M.C.D.) y el mínimo común múltiplo (m.c.m.) de diferentes números, especialmente cuando aparecen quebrados.

El M.C.D. es el valor que es divisor, sin resto, de dos números sin que haya ningún otro número mayor que cumpla esta condición.

El m.c.m. es el número divisible entre los dos números de partida no habiendo otro número más pequeño para el que se cumpla esto.

Si tomamos por ejemplo los dos números Z1=30 y Z2=40, el M.C.D. es 10 y el m.c.m. es 120. (El lector lo puede comprobar mediante cálculo mental).

2. PASO : Análisis del problema

¿Cómo hallar el M.C.D. y el m.c.m. de dos números dados?

Analizemos por medio de los números 30 y 40 cuál es la forma de proceder más adecuada para hallar el M.C.D.:

1. Paso : 40 : 30 = 1 resto 10

2. Paso : 30 : 10 = 3 resto 0

En este ejemplo se halla el M.C.D. de la forma siguiente:

1. Paso : División del número mayor entre el número menor; anotación del resultado.

2. Paso : División del número menor entre el resto; anotación del nuevo resto.

Si el nuevo resto es 0, el divisor que nos ha llevado al mismo es el M.C.D. buscado

Veámos otro ejemplo ilustrativo:

Campo 1 : Z1 (Número mayor)	Campo 2 : Z2 (Número menor)	División	Resto
78	42	1	36
42	36	1	6
36	6	6	0

El M.C.D. de 78 y 42 es 6.

El esquema de cálculo puede transformarse fácilmente en un diagrama de flujo tal como muestra la tabla de arriba.

Pero antes pasemos al mínimo común múltiplo (m.c.m.):

El m.c.m. de dos números Z1 y Z2 se obtiene dividiendo el producto de ambos entre el M.C.D.

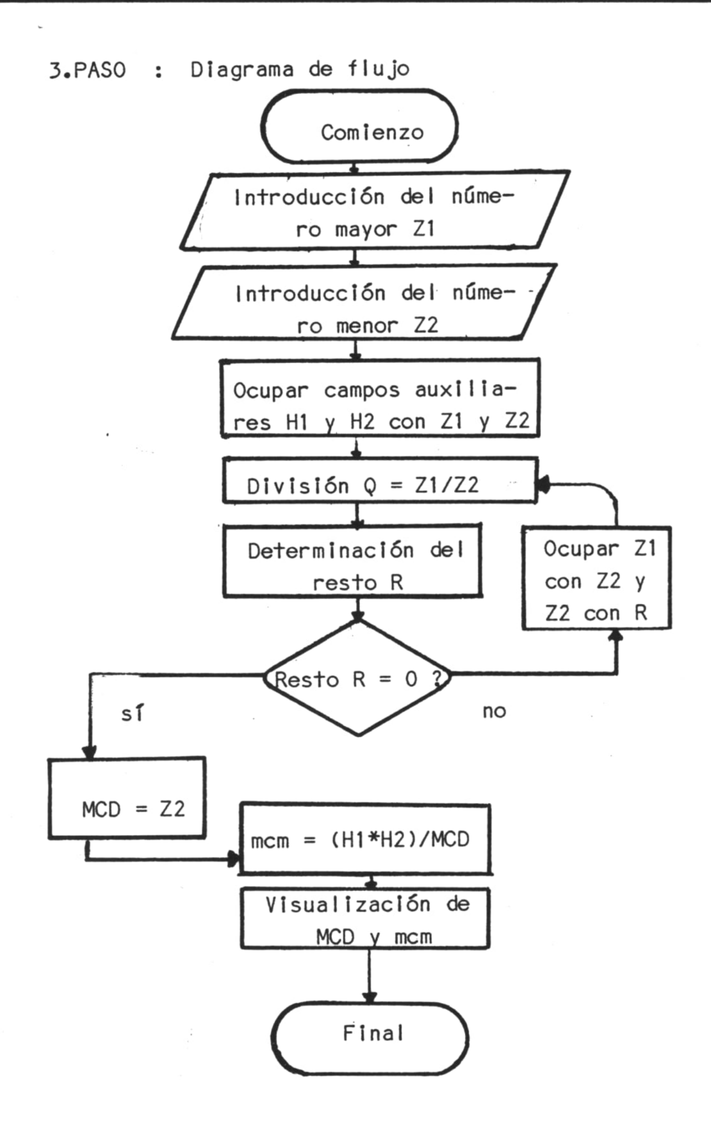
Así, para los números Z1=78 y Z2=42 se obtiene:

$$m.c.m. = \frac{Z1 * Z2}{M.C.D.} = \frac{78 * 42}{6} = 546$$

Para efectuar este cálculo dentro de un programa BASIC según la tabla de arriba cambiando contínuamente el valor

que ocupa cada campo, calcular el M.C.D. e introducir los valores Z1 y Z2 en dos campos auxiliares H1 y H2 de ''memoria intermedia'' para que estén disponibles para el cálculo del m.c.m.

3.PASO : Diagrama de flujo



4. Paso : Programa

```
  10 REM    m.c.d. y m.c.m.
  20 BORDER 0: PAPER 0: INK 7: CLS
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA DETERMINA
R EL MAXI"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"MO COMUN DIVISOR Y EL M
INIMO CO-"
  50 PRINT  PAPER 4; INK 0; FLASH 1;"MUN MULTIPLO DE DOS NUM
EROS.     "
  60 PRINT : PRINT "    -----------------------   "
  70 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  80 PRINT "    -----------------------   "
  90 INPUT "NUMERO MAYOR   -  ?  ";Z1
 100 PRINT AT 9,5;"NUMERO MAYOR    -  ";Z1
 110 INPUT "NUMERO MENOR   -  ?  ";Z2
 120 PRINT AT 11,5;"NUMERO MENOR    -  ";Z2
 130 LET H1=Z1: LET H2=Z2
 140 LET Q=Z1/Z2
 150 LET R=Z1-Z2*INT (Q)
 160 IF R>0 THEN  LET Z1=Z2: LET Z2=R: GO TO 140
 170 LET MCD=Z2
 180 LET MCM=(H1*H2)/MCD
 182 FOR N=0 TO 10
 184 BEEP .05,N
 186 NEXT N
 190 PRINT AT 16,6; PAPER 1; INK 6;"M.C.D.  -  ";MCD
 200 PRINT AT 18,6; PAPER 1; INK 6;"M.C.M.  -  ";MCM
 210 PAUSE 40: PRINT #0; FLASH 1; PAPER 6; INK 4;"        FI
N DEL PROGRAMA          "
 220 PAUSE 200
```

5. Paso : Lista de variables

MCD : Máximo común divisor
H1 : Campo auxiliar 1
H2 : Campo auxiliar 2
MCM : Mínimo común múltiplo
Q : Cociente de los números Z1 y Z2
R : Resto de la división
Z1 : Número mayor
Z2 : Número menor

6. PASO : Descripción del programa

Línea 10-80 : Título y explicaciones

Línea 90-100 : Introducción de un número

Línea 110-120 : Introducción del otro número

Línea 130 : Ocupación de los dos campos auxiliares H1 y H2 con los valores introducidos, para más tarde seguir disponiendo de los mismos (ver 180)

Línea 140 : Determinación del cociente de los dos números introducidos

Línea 150 : Determinación del resto de la división utilizando la función INT

Línea 160 : Si este resto es mayor que 0 se ocupa el campo Z1 con el segundo número Z2 y el campo Z2 con el resto R, volviendo el programa a la línea 140

Línea 170 : Si en cambio el resto es igual a cero, en el campo Z2 tendremos el M.C.D., que será transferido al campo M.C.D.

Línea 180 : De acuerdo con nuestra reflexión,el mcm resulta del producto de ambos números (que aún se encuentran en los campos H1 y H2), dividido entre el MCD.

Línea 182-186 : Bucle para producir un efecto sonoro

Línea 190-200 : Impresión de los resultados con una separación de algunas líneas en blanco

Línea 210-220 : Finalización del programa

7. PASO : Resultados del programa

Después del comando RUN, el programa indica, en una ''pantalla limpia'', lo siguiente :

```
PROGRAMA PARA DETERMINAR EL MAXI
MO COMUN DIVISOR Y EL MINIMO CO-
MUN MULTIPLO DE DOS NUMEROS.
     ---------------------
     PROF. DR.W.VOSS, 1984
     ---------------------

       NUMERO MAYOR  : ?
```

Cuando aparezca el interrogante, que como ya se sabe, indica una interrupción del programa, introducir por ejemplo el

valor 78; a continuación el programa pedirá el segundo número:

NUMERO MENOR : ?

Introduzcamos por ejemplo el número 42 y el ordenador llevará a pantalla los resultados siguientes:

M.C.D. = 6

M.C.M. = 546

FIN DEL PROGRAMA

2.4 Comprobación de números primos

1. PASO : Presentación del programa

Se trata de desarrollar un programa que compruebe si es primo o no un número entero positivo cualquiera que hemos introducido.

Como ya es sabido, los números primos sólo son divisibles por 1 y por sí mismos.

2. PASO : Análisis del problema

Para comprobar si un número dado es primo, hay que dividirlo entre todos los divisores mayores que 1, pero menores que él mismo. Mientras aparezca un resto en estas divisiones o el cociente no sea entero, habrá que seguir dividiendo por otro divisor.

De obtener en una de estas divisiones un resultado sin resto, el número a examen no será primo.

Si, en cambio, se obtiene un resto en todas las divisiones posibles, se llegará a la conclusión de que el número inicial es primo.

Los divisores que deben ser probados comienzan por el 2 y pueden finalizar a la mitad del número inicial; los divisores mayores que éste no es necesario probarlos, porque en ningún caso darán un cociente entero.

Ejemplo : Número inicial X = 25

```
1. Divisor : 2   25/2 = 12 resto 1
2. Divisor : 3   25/3 =  8 resto 1
3. Divisor : 4   25/4 =  6 resto 1
4. Divisor : 5   25/5 =  5 resto 0
                                ===
```

25 no es ningún número primo.

Ejemplo : Número de partida X = 7

1. Divisor : 2   7/2 = 3 resto 1
2. Divisor : 3   7/3 = 2 resto 1

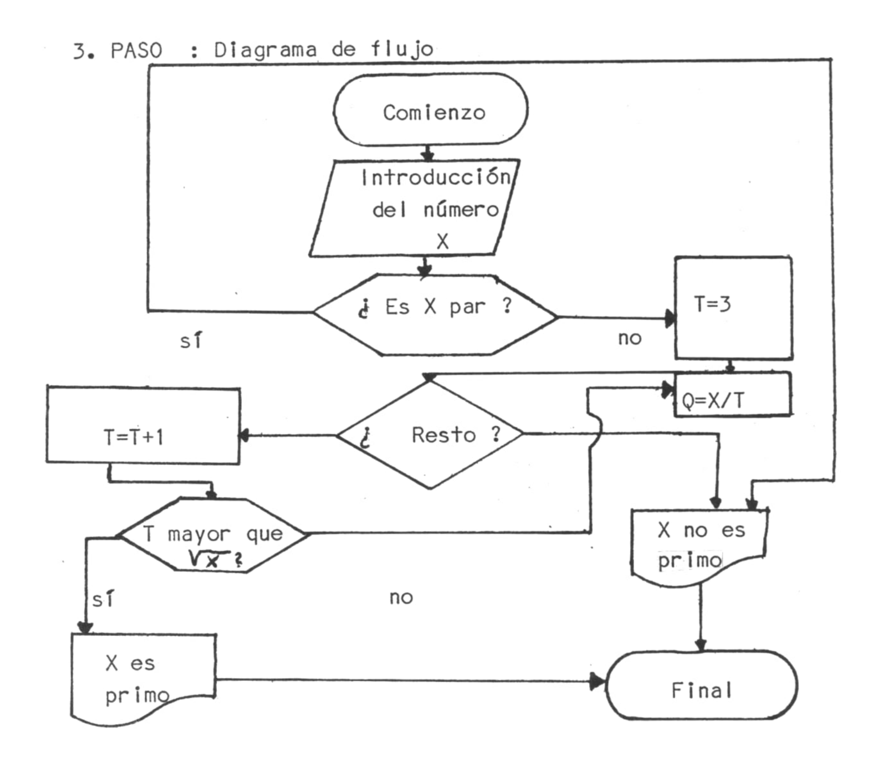
No es necesario probar más divisores, porque el resultado de la división ya no puede ser entero.

Como en este ejemplo no aparece ningún resto 0, podemos decir que:

7 es un número primo.

Estos ejemplos muestran que hay que aumentar paso a paso el divisor comenzando por el valor 2 hasta alcanzar como máximo el valor X/2.

3. PASO : Diagrama de flujo



Este ejemplo muestra como puede simplificarse de modo considerable -y con ello acelerarse- el programa si analizamos cuidadosamente el problema:

Un número par nunca puede ser primo pues, en todo caso, éste ya es divisible por 2. Por ello pueden excluirse los números pares como ''números no - primos'' desde el comienzo del programa.

Nos quedan los números impares. Para un número cualquiera X no es necesario que el divisor llegue a ser X/2; basta con que probemos divisiones hasta máximo $\sqrt{X}$. Si hasta este punto no es posible ninguna división sin resto, tampoco lo será con divisores mayores.

Invitamos al lector a que intente demostrar esta afirmación.

4. Paso : Programa.

```
  10 REM    NUMEROS PRIMOS
  20 BORDER 0: PAPER 0: INK 7: CLS
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA  DETERMI
NAR SI EL"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"NUMERO INTRODUCIDO ES P
RIMO O NO"
  45 PRINT : PRINT : PRINT "    ------------------------   "
  47 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  50 PRINT "    ------------------------  "
  60 INPUT "UN NUMERO ENTERO POR FAVOR   ";X
  70 LET T=2
  75 IF X=0 THEN  GO TO 20
  80 IF X/T=INT (X/T) THEN  GO TO 140
  90 LET Q=X/T
 100 IF Q=INT (Q) THEN  GO TO 140
 110 LET T=T+1
 120 IF T<=SQR (X) THEN  GO TO 90
 130 PRINT AT 15,6;X;" ES UN NUMERO PRIMO": GO TO 150
 140 PRINT AT 15,0;X;" NO ES UN NUMERO PRIMO,  SINO QUE ES D
IVISIBLE ENTRE  ";T
 150 PAUSE 40: PRINT #0; FLASH 1; PAPER 6; INK 4;"        FI
N DEL PROGRAMA         "
 160 PAUSE 200
```

5. Paso : Lista de variables

Q = Cociente entre el número X y el divisor T
T = Divisor
X = Número a introducir para comprobar si es
primo o no.

6. PASO : Descripción del programa

Línea 10-50 : Título, limpieza de la pantalla e impresión del título.

Línea 60 : Demanda del número a comprobar.

Línea 70 : Ocupación del campo divisor con el 2.

Línea 80 : Si X es divisible por 2 (entonces X/2 = INT (X/2)), pasar a la línea 140; si no es así,proseguir en la línea 90.

Línea 90 : División de X entre T (se obtiene Q).

Línea 100 : Si en una de las divisiones el resto es 0 (Q = INT(Q)), seguir en la línea 140, en caso contrario seguir en la línea 110.

Línea 110 : Aumentar el divisor T en 1.

Línea 120 : Mientras T siga siendo más pequeño que $\sqrt{X}$,volver a la línea 90, de lo contrario proseguir en 130.

Línea 130 : Si se alcanza esta línea es porque ninguna de las divisiones ha dado resto 0. Esto significa que X debe ser número primo.

Se imprime el resultado y tiene lugar un salto a la línea 150.

Línea 140 : La línea 140 se alcanza únicamente en el caso de haber obtenido resto 0 en una de las divisiones.
Esto supone que X no es primo, y que es (como mínimo) divisible por el valor que hay en el campo T. Esto se visualiza.

Línea 150 : Finalización del programa.

Línea 160 : Temporizador (200/50=4 segundos)

7. PASO : Resultados

En este y en próximos ejemplos prescindiremos de repetir la impresión del título.

Después de aparecer el título, el programa pide el número a comprobar y lo almacena en el campo X (también prescindiremos en este y en próximos ejemplos de reproducir los correspondientes textos de demanda).

Introduzcamos p.e. el número 13 y el ordenador contestará:

```
13 ES UN NUMERO PRIMO

   FIN DEL PROGRAMA
```

Si en cambio introducimos el número 27, el ordenador nos contestará:

```
27 NO ES UN NUMERO PRIMO, SINO
   QUE ES DIVISIBLE POR 3


        FIN DEL PROGRAMA
```

2.5 Ecuación de segundo grado

1. PASO : Presentación del problema

La resolución de ecuaciones de segundo grado se cuenta entre los problemas fundamentales de la matemática en los últimos niveles de EGB y primeros del BUP y prepara para problemas de mayor complejidad.

La forma general de una ecuación de segundo grado en la cual hay que despejar la incógnita x es:

$$(1) \qquad ax^2 + bx + c = 0$$

Dividiendo esta igualdad por el coeficiente a, se obtiene:

$$(2) \qquad x^2 + px + q = 0$$

Se obtienen dos soluciones x y x a partir de la fórmula siguiente:

$$(3) \qquad x = -p/2 \pm \sqrt{p/4 - q}$$

Según la constelación de coeficientes a, b y c, la fórmula (3) nos llevará a resultados distintos.

Por ejemplo, cuando el valor bajo la raiz es negativo, obtendremos resultados fuera del campo de los números reales. Este valor se llama discriminante(D).

Se obtienen también soluciones especiales cuando el coeficiente a es igual a 0;en este caso ya no tenemos una ecuación de segundo grado.

Por ello, de ahora en adelante contemplaremos sólo los casos ''auténticos'' ($a \neq 0$) y al mismo tiempo con soluciones reales (D>0). En todos los demás casos el programa deberá generar un mensaje de interrupción.

Para resolver el problema utilizaremos la fórmula (3) en su variante (4):

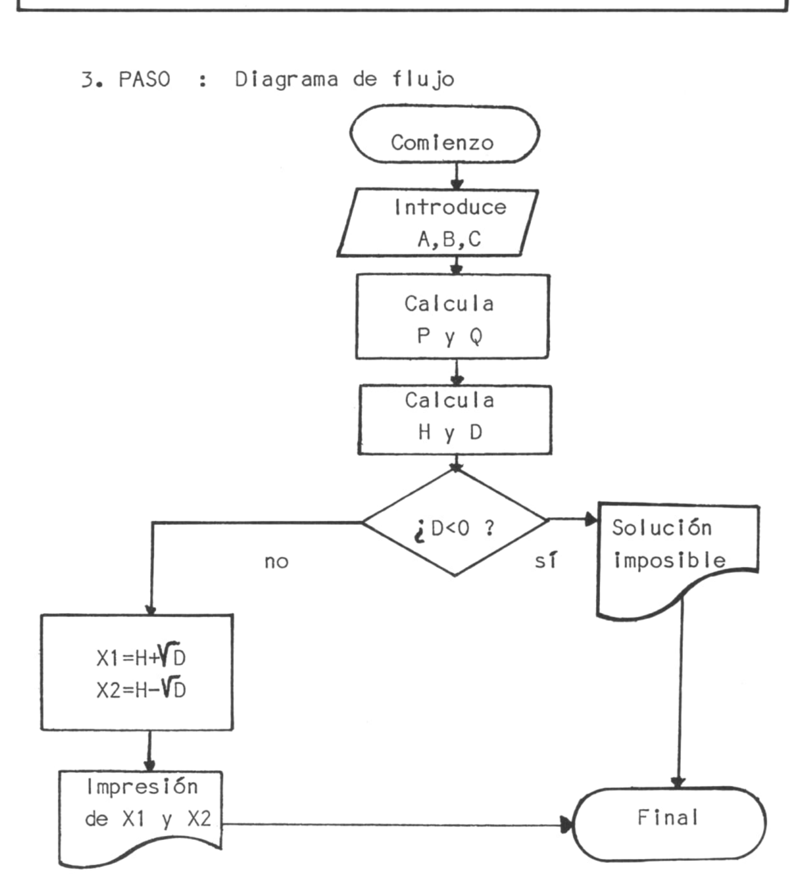
(3) $$x = -p/2 \pm \sqrt{p/4 - q}$$

(4) $$x = H \pm \sqrt{D}$$

H es un valor auxiliar que equivale a -p/2

El correspondiente programa BASIC debe determinar primero los valores H y D, para luego imprimir las soluciones o un mensaje de interrupción según sea D>0 o D<0.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
 10 REM  E. DE SEGUNDO GRADO
 20 BORDER 0: PAPER 0: INK 7: CLS
 30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA RESOLVER
UNA ECUA-"
 40 PRINT  PAPER 4; INK 0; FLASH 1;"CION DE SEGUNDO GRADO.
 "
 45 PRINT : PRINT "    ------------------------   "
 47 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
 50 PRINT "     ------------------------   "
 60 INPUT "COEF. TERMINO CUADRATICO A ? ";A
 70 INPUT "COE. DEL TERMINO LINEAL  B ? ";B
 80 INPUT "TERMINO INDEPENDIENTE   C ? ";C
 90 LET P=B/A: LET Q=C/A
100 LET H=P/2: LET D=P*P/4-Q
110 IF D<0 THEN  BEEP 2,-7: PRINT AT 15,7;"SOLUCION IMPOSIB
LE": GO TO 150
120 LET X1=H+SQR (D): LET X2=H-SQR (D)
125 PRINT AT 10,11; FLASH 1;"SOLUCIONES"
130 PRINT AT 15,11;"X1 = ";X1
140 PRINT AT 17,11;"X2 = ";X2
150 PAUSE 40: PRINT #0; FLASH 1; PAPER 6; INK 4;"        FI
N DEL PROGRAMA        "
160 PAUSE 200
```

5. Paso : Lista de variables

A = Coeficiente de la parte cuadrática.
B = Coeficiente de la parte lineal
C = Constante de la ecuación cuadrática
D = Discriminante
P = B/A
H = -P/2
Q = C/A
X1= Primera solución.
X2= Segunda solución.

6. PASO : Descripción del programa

Línea 10-50 : Comentario e impresión del título

Línea 60-80 : Introducción de los coeficientes de la igualdad cuadrática (en la forma (1))

Línea 90 : Determinación de P y de Q según la igualdad (2)

Línea 100 : Determinación del valor auxiliar H y del discriminante D

Línea 110 : Si D es menor que 0, tiene lugar un mensaje de interrupción y se pasa a la línea 150

Línea 120 : Si, en cambio, el discriminante no es negativo, se calculan las dos soluciones X1 y X2

Línea 130-140: Ambas soluciones se visualizan en la pantalla

Línea 150 : Final del programa

7. PASO : Resultados

Introduzcamos p.e.:

A = 4,
B = 4,
C = -24,

¡El lector debería probar también otros valores!

Se trata de la ecuación de segundo grado:

$$4x^2 + 4x - 24 = 0,$$

tras lo cual el programa visualiza los siguientes resultados:

```
X1 = 2
X2 = -3

FIN DEL PROGRAMA
```

2.6 Número e

1. PASO : Presentación del problema

El número de Euler, abreviado e, llamado también constante de crecimiento, juega un papel importante en el desarrollo matemático de series, que pueden ser contempladas como cadenas de crecimiento o de desarrollo.

Este sirve además como base de los llamados logarítmos neperianos, que volveremos a encontrarnos en el capítulo dedicado a problemas de economía.

Si nos hemos olvidado del valor del número e, podemos calcularlo con ayuda de un programa BASIC adecuado y siempre sabiendo cómo surge e.

Se trata del límite de la siguiente sucesión, cuando n tiende hacia infinito:

$e = (1 + 1/n)$ para n hacia infinito.

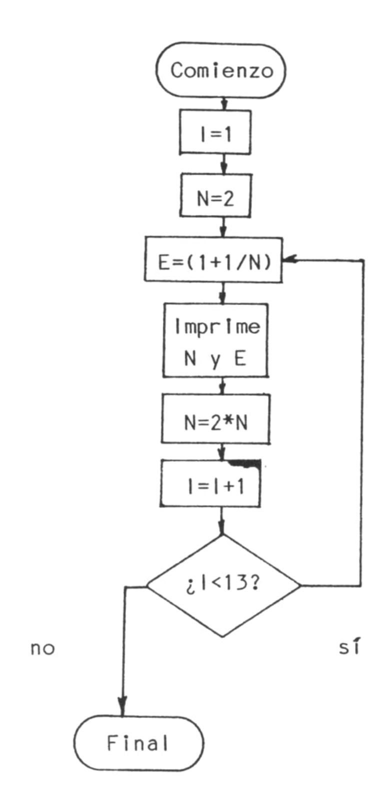
2. PASO : Análisis del problema

El análisis del problema es muy sencillo en este caso, dado que se trata únicamente de programar la fórmula anterior.

Este programa puede probarse con valores crecientes y alternativos de n. Como el valor e se alcanza sólo con n tendiendo hacia infinito, recomendamos hacer crecer rápidamente n, por ejemplo doblando su valor en cada cálculo.

Ello exige, no obstante, empezar con un valor de n mayor que 1 (es decir, p.e. n = 2), para que n no se quede estancado en 1 al duplicarlo.

3. PASO : Diagrama de flujo



4. Paso : Programa

```
  10 REM        EULER
  20 BORDER 0: PAPER 0: INK 7: CLS
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA DETERMINA
R EL NUME"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"RO E.
       "
  45 PRINT : PRINT "    ------------------------  "
  47 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  50 PRINT "    ------------------------  "
  60 LET N=2
  70 PRINT : PRINT "       N","        E": PRINT "----------
----------------------": PRINT
  80 FOR I=1 TO 12
  90 LET E=(1+1/N)^N
 100 PRINT "   ";N,"     ";E
 110 LET N=N*2
 120 NEXT I
 130 PAUSE 40: PRINT #0; FLASH 1; PAPER 6; INK 4;"        FI
N DEL PROGRAMA        "
 140 PAUSE 200
```

5. Paso : Lista de variables

E = Número e (valor aproximado)
I = Indice contador de bucle
N = Parámetro en la fórmula para el cálculo del número e

6. PASO : Descripción del programa

Línea 10-50 : Título

Línea 60 : Ocupación del parámetro N con el valor inicial 2

Línea 70 : Impresión del título de la tabla y de una línea en blanco

Línea 80-120 : Recorrer 12 veces el bucle. En cada vuelta se calcula E, se imprimen N y E y se duplica N

Línea 130 : Final del programa

7. PASO : Resultados

Este programa genera una tabla, en la que se enfrentan los valores crecientes de N con los valores resultantes de E.

En esta tabla puede observarse como e se acerca a un valor fijo.

Para acercarnos aún más al límite, deberíamos recorrer más vueltas de bucle.

Entonces habría que modificar la línea 80, por ejemplo del modo siguiente:

```
80 FOR I = 1 TO 30
```

En todo caso, hay que tener en cuenta que los valores demasiado altos de n pueden conducir a errores, debido a las imprecisiones de cálculo del ordenador.

La tabla resultante del programa anterior ofrece este aspecto:

n	e
2	2.25
4	2.4414063
8	2.5657845
16	2.6379285
32	2.6769901
64	2.697345
128	2.707739
256	2.7129916
512	2.715632
1024	2.7169557
2048	2.7176185
4096	2.7179501

FIN DEL PROGRAMA

2.7 Cálculo de porcentajes

1. PASO : Presentación del problema

La regla de tres juega un papel fundamental en el cálculo de porcentajes. Por ejemplo responde a la pregunta de qué parte de una cantidad total G representa un determinado importe B. Podemos ilustrar este problema con un ejemplo típico:

El importe de una factura asciende a Ptas. 45000.—. El cliente debe pagar adicionalmente el llamado impuesto sobre el valor añadido, que supone un 10% del importe neto de la factura, de lo cual resulta:

Importe bruto = Importe neto + Impuesto

El problema que aquí se plantea consiste entonces en determinar, primero el importe del impuesto y después el importe bruto, para un importe neto y un porcentaje del impuesto dados.

Naturalmente podría pedirse el importe neto sabiendo el porcentaje y el importe bruto.

2. PASO : Análisis del problema

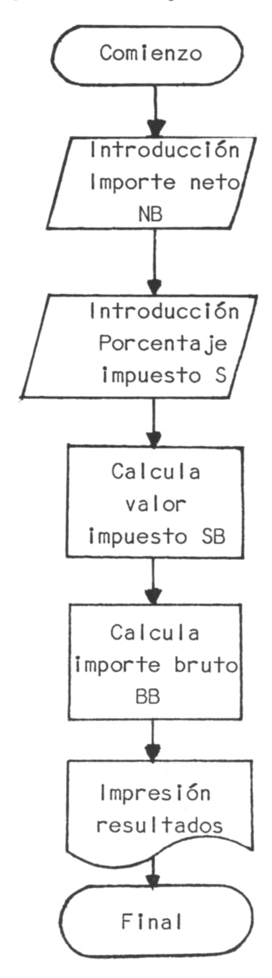
El ejemplo citado arriba puede solucionarse por regla de tres del modo siguiente (señalándonos así el modo de resolución):

```
Importe neto = 100 % equivalente a 45000.- Ptas.
Impuesto     = 14 % equivalente a    x     Ptas.
-------------------------------------------------

                 45000*10
           x =   --------   =  4500.- (Ptas.)
                   100
```

Con ello queda claro el camino a seguir: Se trata de introducir en un programa general el importe neto y el porcentaje del impuesto (o de forma más general, el porcentaje, que puede elegirse arbitrariamente para cualquier aplicación).

3. PASO : Diagrama de flujo



4. Paso : Programa

```
  10 REM     PORCENTAJE
  20 BORDER 0: PAPER 0: INK 7: CLS
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA ESTIMACIO
N DE POR-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"CENTAJES.
       "
  45 PRINT : PRINT "     -------------------------   "
  47 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  50 PRINT "    -------------------------   "
  60 INPUT "IMPORTE NETO ?  ";NB
  70 INPUT "PORCENTAJE   ?  ";SS
  80 LET SB=NB*SS/100
  90 LET BB=NB+SB
 100 PRINT AT 11,5;"IMPORTE NETO     = ";NB
 110 PRINT AT 13,5;"VALOR PORCENTUAL = ";SB;"%"
 120 PRINT AT 15,5;"IMPORTE BRUTO    = ";BB
 130 PAUSE 40: PRINT #0; FLASH 1; PAPER 6; INK 4;"        FI
N DEL PROGRAMA        "
 140 PAUSE 200
```

5. Paso : Lista de variables

BB = Importe bruto
NB = Importe neto
SB = Valor porcentual
SS = Porcentaje

6. PASO : Descripción del programa

Línea 10-50 : Título.

Línea 60-70 : Introducción del importe neto y del porcentaje.

Línea 80-90 : Cálculo del importe del porcentaje y del importe bruto.

Línea 100-120 : Impresión en pantalla de los resultados

Línea 130 : Final del programa.

7. PASO : Resultados

Si se introduce por ejemplo el importe neto 200 y el porcentaje 15%, el programa contestará:

```
IMPORTE NETO       :  200
VALOR PORCENTUAL   :  30
IMPORTE BRUTO      :  230

     FIN DEL PROGRAMA
```

2.8 Dado

1. PASO : Presentación del problema

Presentamos un programa que "echa" 100 veces los dados y que calcula el promedio de puntos obtenido.

Para este fin puede hacerse uso de la función RND; (RND = random = azar), que genera uno de los 65536 numeros que forman la secuencia de números aleatorios.

Números aleatorios son aquéllos que aparecen de forma irregular y no sistemática.

El valor retornado por RND está comprendido entre 0 y 1, en algunas ocasiones puede valer 0, pero nunca 1.

2. PASO : Análisis del problema

Para conseguir números aleatorios comprendidos entre otros límites, podríamos utilizar la función RND * N, donde N es un número cualquiera. Esto hará que el ordenador retorne una secuencia de números aleatoria comprendida entre 0 y N-1. Si se desean obtener números aleatorios enteros, deberá escribirse INT (RND * N). Si lo que deseamos es que el ordenador retorne números aleatorios enteros conprendidos entre dos cualesquiera, ambos inclusive, escribiremos: INT ((X-Y+1) * RND) + Y, donde X es el número mayor e Y el menor.

El lector debería probar simplemente esta función, introduciendo por ejemplo:

```
PRINT RND
```

y otra vez:

```
PRINT RND
```

o todavía mejor:

```
FOR I = 1 TO 10 : PRINT RND: NEXT I
```

y seguidamente la tecla ENTER.

En la pantalla aparecerán números aleatorios situados todos ellos entre 0 y 1.

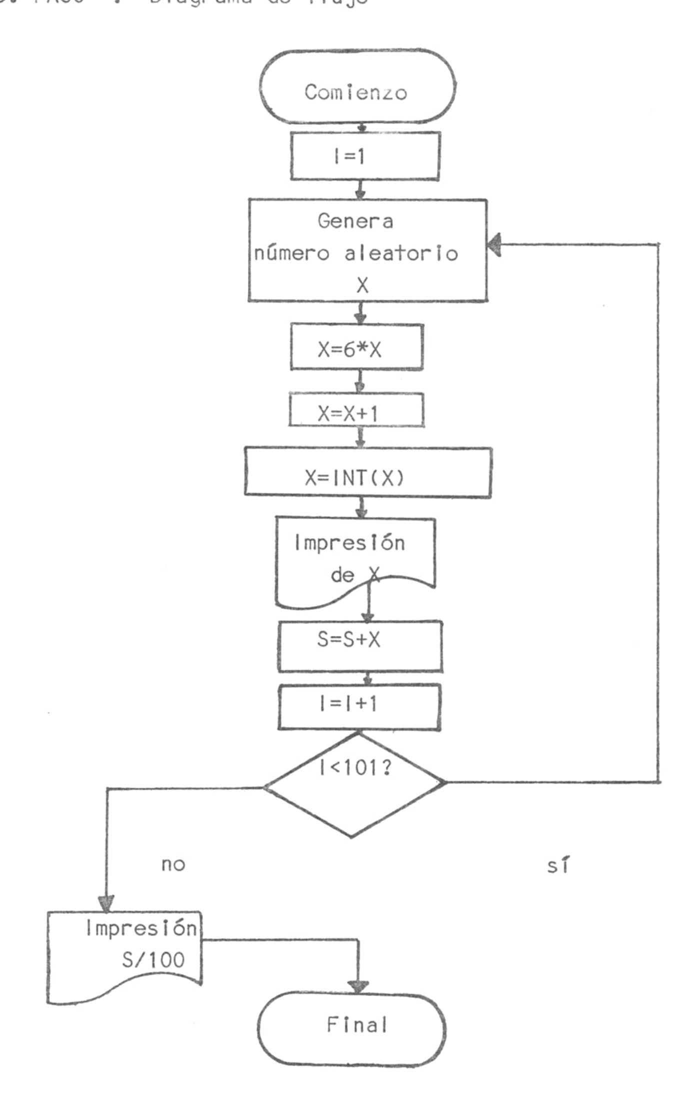
No obstante, queremos obtener como resultado 100 valores enteros situados entre 1 y 6, como cuando se echa un dado

El esquema siguiente ilustra cómo pueden generarse números enteros entre el 1 y el 6 a partir de números aleatorios tomados del ámbito de valores entre 0 y 1:

Elaboración	Resultado
Función RND	Entre 0 y menos de 1
* 6	entre 0 y menos de 6
+ 1	entre 1 y menos de 7
Función INT	entero entre 1 y 6

Con ello se ha trazado ya el camino para resolver el problema. Como finalmente se desea obtener el promedio, habrá que sumar todos los resultados y dividir esta suma entre 100.

3. PASO : Diagrama de flujo



4. Paso : Programa

```
  10 REM        DADO
  20 BORDER 0: PAPER 0: INK 7: CLS
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA GENERAR
100 TIRA-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"DAS DE DADO Y CALCULAR
EL PROME-"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"DIO DE PUNTOS.
       "
  50 PRINT : PRINT "    ------------------------  "
  60 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  70 PRINT "    ------------------------  ": PRINT : PRINT
  80 LET I=1: LET S=0
  90 LET X=RND
 100 LET X=X*6: LET X=X+1: LET X=INT (X)
 110 PRINT X;" ";
 120 LET S=S+X
 130 LET I=I+1
 140 IF I<101 THEN  GO TO 90
 150 LET AM=S/100
 160 FOR N=12 TO 18: BEEP .2,N: NEXT N
 170 PRINT AT 19,0;"PROMEDIO DE LAS TIRADAS = ";AM
 180 PAUSE 40: PRINT #0; FLASH 1; PAPER 6; INK 4;"        FI
N DEL PROGRAMA        ": PAUSE 200
```

5. Paso : Lista de variables

AM = Valor medio
I = Contador de bucle
S = Suma de todas las tiradas
X = Valor aleatorio

6. PASO : Descripción del programa

Línea 10-70 : Título

Línea 80 : Ajuste del contador de vueltas

Línea 90-100 : Generación de un número aleatorio y transformación en un número entero situado entre 1 y 6

Línea 110 : Impresión

Línea 120 : Incremento de la suma S en el resultado de la tirada X

Línea 130 : Salto de uno del contador de vueltas

Línea 140 : Mientras I sea menor que 101 vuelta a la línea 90

Línea 150-170 : Cálculo del promedio e impresión

Línea 180 : Final del programa

7. PASO : Resultados

Después de iniciar el programa, éste genera tras el título 100 sucesos aleatorios de tirada de un dado y los imprime línea a línea (pensar por qué tiene lugar una impresión línea a línea).
A continuación el ordenador calcula, a partir de los 100 sucesos, el promedio y lo visualiza. Si el generador de números aleatorios funciona correctamente, el resultado deberá acercarse a 3.5.

Como los resultados dependen del azar, lógicamente no los podemos representar aquí.

2.9 Exactitud de los cálculos

El SPECTRUM reserva un número limitado de bytes para cada número real con el que va a calcular. Esto significa que la precisión con la que trabaja está limitada. Puede verse esto claramente si introducimos valores "grandes" en el programa que sigue, haciendo que se impriman a continuación :

```
 10 REM Exactitud de calculo
 20 LET A=123456
 30 LET B=1234567
 40 LET C=12345678
 50 LET D=99999999
 60 LET E=-99999999
 70 LET F=123456789
 80 LET X=1234567899
 90 LET G=0.00001
100 LET H=0.000001
110 PRINT "A=";A'"B=";B'"C=";C'"D=";D'"E=";E'"F=";F'"X=";X
"G=";G'"H=";H'"F+1=";F+1'"F+10=";F+10
```

Obtenemos los siguientes resultados:

```
A=123456
B=1234567
C=12345678
D=99999999
E=-99999999
F=1.2345679E+8
X=1.2345679E+9
G=.00001
H=1E-6
F+1=1.2345679E+8
F+10=1.234568E+8
```

Es decir, si el ordenador recibe números de más de 8 cifras (ejemplo F) a la hora de imprimir, los pasa a la llamada representación exponencial, como por ejemplo para F:

1.2345679E+8

Esto hay que leerlo como:

1.2345679 * 100 000 000

Este producto da como resultado:

123456790

Por consiguiente, el ordenador ha redondeado por encima de la última posición del valor de F, que ya no puede considerar. En todo caso, como puede mostrarse por medio de otros ejemplos, este redondeo se efectúa no ''recortando'' simplemente la posición.
El principal problema que surge aquí es el siguiente:
Si, por ejemplo, sumamos 1 al valor F en el programa anterior, y hacemos que se imprima también el resultado en pantalla, obtendremos:

1.2345679E+8

no advertiremos ninguna diferencia entre los valores de F y de F+1 - ¡el ordenador calcula mal! -
Este problema deberá ser considerado siempre que queramos operar números muy grandes con números muy pequeños (o viceversa).
El campo de notación entera está comprendido entre +99999999 y -999999999. En notación decimal entre +0.00001 y +99999999 / -0.00001 y -99999999 y en notación exponencial entre 1.7014118E+38 y 2.9387359E-39.



Capítulo 3 : Química

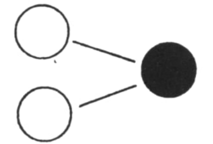
3.1 Otras instrucciones BASIC

Para confeccionar programas en el campo de la química pueden aprovecharse magníficamente las posibilidades gráficas y de color de los ordenadores SPECTRUM - por ejemplo para representar gráficamente moléculas o enlaces químicos -.

Tomemos el caso de la molécula de agua:

H_2O

que, como ya es sabido, está compuesta de dos átomos de hidrógeno (H) y de un átomo de oxígeno (O), representables de la forma siguiente:



$H_2 - O$

Para generar representaciones como ésta se necesitan algunas instrucciones adicionales. Estas se refieren a la utilización y creación de los llamados gráficos definidos por el usuario - GDU -, para lo cual se incluye un programa en la página 97 del libro.

Para confeccionar este tipo de dibujos resulta muy adecuado el juego de caracteres gráficos que puede definir el usuario con la ayuda del mencionado programa.
Además el SPECTRUM viene de fábrica con una serie de gráficos predefinidos, situados en la fila de teclas del 1 al 8. Para acceder a todos estos caracteres gráficos debemos pasar a modo gráfico. Esto se consigue pulsando a la vez CAPS SHIFT + 9, de este modo, el cursor pasa a mostrar una G parpadeante.
Estos caracteres, que pueden hallarse bajo determinados códigos ASCII pueden seleccionarse con la función:

CHR$ (argumento)

El argumento de esta función es el número de código ASCII del símbolo deseado.

Para visualizar todos los símbolos del código ASCII puede utilizarse el programa siguiente:

```
 10 REM TABLA ASCII DEL SPECTRUM
 20 FOR I=0 TO 5
 30 PRINT I, FLASH 1; PAPER 4; INK 2;"CODIGO NO USADO"
 40 NEXT I
 50 PRINT 6,"PRINT COMA"
 60 PRINT 7,"EDIT"
 70 PRINT 8,"CURSOR IZQUIERDA"
 80 PRINT 9,"CURSOR DERECHA"
 90 PRINT 10,"CURSOR ABAJO"
100 PRINT 11,"CURSOR ARRIBA"
110 PRINT 12,"DELETE"
120 PRINT 13,"ENTER"
130 PRINT 14,"numero"
140 PRINT 15, FLASH 1; PAPER 4; INK 2;"CODIGO NO USADO"
150 PRINT 16,"INK control"
160 PRINT 17,"PAPER control"
170 PRINT 18,"FLASH control"
180 PRINT 19,"BRIGHT control"
190 PRINT 20,"INVERSE control"
200 PRINT 21,"OVER control"
210 PRINT 22,"AT control"
220 PRINT 23,"TAB control"
230 FOR I=24 TO 31
240 PRINT I, FLASH 1; PAPER 4; INK 2;"CODIGO NO USADO"
250 NEXT I
260 PRINT 32,"ESPACIO"
270 INK 6
280 FOR I=33 TO 255
290 IF I>47 THEN  INK 5
300 IF I>57 THEN  INK 4
310 IF I>64 THEN  INK 2
330 IF I>127 AND I<165 THEN  INK 3: FOR X=128 TO 164: PRINT
X,"GRAPHICS ";CHR$ X: NEXT X: LET I=X
340 IF I>164 THEN  INK 6
350 PRINT I,CHR$ I
360 NEXT I
```

```
   1 REM
   5 REM CARGADOR DE CARACTERES GRAFICOS PREDEFINIDOS (UDG)
   6 REM
  10 FOR n-65368 TO 65375
  20 READ x
  30 POKE n,x
  40 NEXT n
  60 DATA BIN 00011000
  70 DATA BIN 01111110
  80 DATA BIN 01111110
  90 DATA BIN 11111111
 100 DATA BIN 11111111
 110 DATA BIN 01111110
 120 DATA BIN 01111110
 130 DATA BIN 00011000
 200 FOR n=USR "b" TO USR "c"-1
 210 READ x
 220 POKE n,x
 230 NEXT n
 240 DATA BIN 10000000
 250 DATA BIN 01000000
 260 DATA BIN 00100000
 270 DATA BIN 00010000
 280 DATA BIN 00001000
 290 DATA BIN 00000100
 300 DATA BIN 00000010
 310 DATA BIN 00000001
 400 FOR n=USR "c" TO USR "d"-1
 410 READ x
 420 POKE n,x
 430 NEXT n
 440 DATA BIN 00000001
 450 DATA BIN 00000010
 460 DATA BIN 00000100
 470 DATA BIN 00001000
 480 DATA BIN 00010000
 490 DATA BIN 00100000
 500 DATA BIN 01000000
 510 DATA BIN 10000000
 600 FOR n=USR "d" TO USR "e"-1
 610 READ x
 620 POKE n,x
 630 NEXT n
 640 DATA BIN 00000000
 650 DATA BIN 00000000
 660 DATA BIN 00000000
 670 DATA BIN 11111111
 680 DATA BIN 00000000
 690 DATA BIN 00000000
 700 DATA BIN 00000000
 710 DATA BIN 00000000
 720 CLS
1000 PRINT AT 0,0; INK 2;"O"
1100 PRINT AT 1,1;"\"
1200 PRINT AT 3,1;"/"
1300 PRINT AT 2,2; INK 6;"O"
1400 PRINT AT 4,0; INK 2;"O"
```

Cada GDU está formado por una matriz de 8x8 píxels, esto se traduce en que se necesitan 8 bytes para definir cada caracter gráfico o GDU. En el SPECTRUM, es posible definir un total de 21 de estos caracteres gráficos especiales. Siempre que se utilicen estos caracteres, será preciso cargar en primer lugar el programa de la página anterior. Haga luego NEW y cargue el programa en que aparecen dichos caracteres gráficos. Los GDU empleados por los programas de este libro son los definidos por el programa anterior.

Este programa mueve línea a línea y por toda la pantalla el GDU A (una bolita blanca).

```
10  REM       -ANIMACION-
20  PAPER 0:BORDER 0:INK 7:CLS
30  FOR X=0 TO 21
40  FOR Y=0 TO 31
50  PRINT AT X,Y;"A"
60  PAUSE 5
70  CLS
80  NEXT X
90  NEXT Y
100 PAUSE 0:STOP
```

Además, en este capítulo, y en la mayoría de los que siguen, necesitaremos ciertas instrucciones que nos permitan introducir mayores cantidades de información de una forma más elegante que con las instrucciones PRINT e INPUT.

Para esto resulta particularmente adecuada la instrucción DATA, en unión de la instrucción READ.

Instrucción 14:

```
nn DATA Valor 1, Valor 2, Valor 3, ...
```

Instrucción 15:

```
nn READ variable 1, variable 2, variable 3,...
```

Ejemplo:

```
10 REM -INTRODUCCION DATOS-
20 PAPER 0:BORDER 0:INK 7:CLS
30 DATA 4600,4630,8000,2000
40 READ P1,P2,P3,P4
50 PRINT "P1 = ";P1,"P2 = ";P2
60 PRINT "P3 = ";P3,"P4 = ";P4
70 STOP
```

Invitamos al lector a probar este programa.

Cuando la cantidad de datos sea elevada, recomendamos utilizar un único nombre de variable que, no obstante, tomará varios valores. Esto se consigue utilizando variables indexadas como las que ya conocemos de las matemáticas. Normalmente en términos matemáticos se escribe:

x_1, x_2, x_3, ... término general x_i

Esto puede llevarse a cabo de la misma forma con el lenguaje BASIC:

X(1), X(2), X(3), ... término general X(I)

(en general:
nombre variable (nombre del índice))

Cuando utilizamos estas variables indexadas hay que comunicar al ordenador al comienzo del programa cuántas posiciones de memoria deben mantenerse libres para cada una de las variables en el principio del programa.

Para ello utilizaremos la instrucción DIM:

Instrucción 16
----

nn DIM nombre var. (num,num,...........)

Así por ejemplo la instrucción:

```
10 DIM X(20)
```

reserva espacio en la memoria del ordenador para 20 elementos, de la matriz unidimendional X.

Aquí se trata de los elementos X(1), X(2), X(3),..., X(20).

Para completar esta explicación añadiremos que son también posibles indexados dobles, triples, cuádruples...:

La instrucción:

```
10 DIM Z(3,4)
```

mantiene libres 3*4 posiciones de memoria para la variable Z, que uno puede imaginarse como una matriz de tres líneas y cuatro columnas:

Z(1,1)	Z(1,2)	Z(1,3)	Z(1,4)
Z(2,1)	Z(2,2)	Z(2,3)	Z(2,4)
Z(3,1)	Z(3,2)	Z(3,3)	Z(3,4)

El formato general es en este caso:

Z(I,J)

El primer índice (I) es el llamado índice de línea y el segundo índice (J) es el llamado índice de columna.

Para finalizar queremos indicar que la instrucción READ de un programa, toma valores de los DATAS en orden sucesivo. Ello significa que una segunda instrucción READ proseguirá con la lectura de los DATAS, allí donde dejó de hacerlo la última instrucción READ.

Aún así, a menudo es más razonable volver a leer desde el principio la lista de DATAS con una segunda (o siguiente) instrucción READ.

Para hacer posible esto necesitamos una instrucción nueva:

Instrucción 17:

```
nn RESTORE  num. de línea
```

Esta instrucción provoca que el ''cursor de lectura'' sea emplazado nuevamente al comienzo de la primera instrucción DATA de la línea especificada, de modo que los datos sean leídos nuevamente desde el principio con la siguiente instrucción READ.

Si se omite el número de línea, el intérprete BASIC toma por defecto el valor 0, con lo que el puntero quedaría posicionado en la primera sentencia DATA del programa.

3.2 La molécula de agua

1. PASO : Presentación del problema

Vamos a generar, con ayuda de un programa BASIC, la representación gráfica de la molécula de agua (H_2O); (véase también el punto 3.1). Los átomos de hidrógeno se representarán p.e. en forma de una bolita azul y el átomo de oxígeno como una bolita blanca.

Así, este programa puede usarse como base de programas más complejos de este tipo de representación gráfica.

2. PASO : Análisis del problema

Una vez tenemos ante nosotros el gráfico que debe generarse con este programa, el análisis del problema resulta muy sencillo:

Se trata de producir exactamente el mismo dibujo que ya realizamos al comienzo del punto 3.1.

Observaremos que este dibujo se compone de cinco líneas:

1. Línea : una bolita azul.

2. Línea : una línea oblícua de arriba a la izquierda a abajo a la derecha.

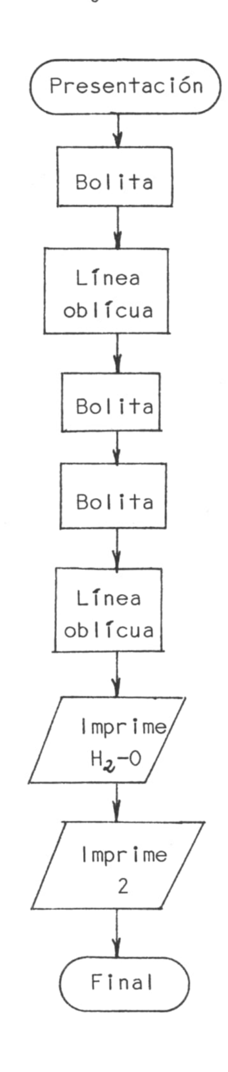
3. Línea : una bolita blanca

4. Línea : una línea oblícua de abajo a la izquierda a arriba a la derecha.

5. Línea : una bolita azul.

Las bolitas y las líneas oblícuas son generadas por el programa de la página 97. Para incorporar estos gráficos a los programas que vienen a continuación debera entrar en modo grafico (pulsar CAPS SHIFT + 9), y pulsar las letras a (bolita), b (línea oblícua de arriba a la izquierda a abajo a la derecha) y c (línea oblícua de abajo a la izquierda a arriba a la derecha).

3. PASO : Diagrama de flujo



4. Paso : Programa

```
 10 REM    QUIMICA I
 20 BORDER 0: PAPER 0: INK 7: CLS
 30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA LA REPRE
SENTACION"
 40 PRINT  PAPER 4; INK 0; FLASH 1;"DE LA MOLECULA DE AGUA
H2-O.    "
 45 PRINT : PRINT "    ------------------------  "
 47 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
 50 PRINT "    ------------------------  "
 60 PRINT AT 12,8; INK 5;"O"
 70 PRINT AT 13,9; INK 6;"\"
 80 PRINT AT 14,10;"O"
 90 PRINT AT 12,12; INK 5;"O"
100 PRINT AT 13,11; INK 6;"/"
110 PRINT AT 13,20;"H -O"
120 PRINT AT 14,21;"2"
130 PRINT #0;"       "; FLASH 1; PAPER 1; INK 6;"  FIN DEL P
ROGRAMA  ": PAUSE 400
```

5. Paso : Lista de variables

En este programa no se utiliza ninguna variable

6. PASO : Descripción del programa

Línea 10-50 : Título del programa.

Línea 60 : Impresión de una bolita azul en la línea 12, columna 8.

Línea 70 : Impresión de una raya oblicua en la línea 13, columna 9.

Línea 80 : Impresión de una bolita blanca en la línea 14, columna 10.

Línea 90 : Impresión de una bolita azul en la línea 12, columna 12.

Línea 100 : Impresión de una raya oblicua en la línea 13, columna 11.

Línea 110 : Impresión de H -O en la línea 13, columna 20.

Línea 120 : Impresión del subíndice 2

Línea 130 : Impresión de "FIN DEL PROGRAMA"

7. PASO : Resultados

La impresión en pantalla de este programa es similar a la que presentamos en el punto 3.1.

3.3 Reacción química

1. PASO : Presentación del problema

Las reacciones químicas también pueden representarse gráficamente de una forma clara, como se demuestra en el ejemplo del proceso de fabricación del acero.

La fabricación del acero aparece en los países industrializados con la producción de hierro bruto en los altos hornos. Esta producción se desarrolla en los siguientes pasos:

1. Paso : Se quema aire caliente con carbón para obtener monóxido de carbono:

$$2\,C + O_2 \longrightarrow 2\,CO$$

carbón oxígeno monóxido de carbono

2. Paso : El monóxido de carbono (CO) caliente asciende por el horno y reacciona allí con el óxido férrico (Fe 0):

Fe 0 + 3 CO ——→ 2 Fe + 3 CO

Oxido férrico	Monóxido de carbono	Hierro	Dióxido de carbono

De este modo se obtiene hierro bruto para su posterior utilización.

Presentamos ahora un programa BASIC que sirve para mostrar estas reacciones químicas.

Para la representación de los átomos y de las moléculas nos serviremos de las mismas posibilidades gráficas que en el ejemplo precedente.

2. PASO : Análisis del problema

En realidad, con la descripción anterior y considerando lo dicho en el ejemplo anterior, el análisis del problema ya queda resuelto, de modo que podemos ahorrarnos cualquier otra explicación.

3. Paso : Diagrama de flujo

Tampoco aquí resulta necesario el diagrama de flujo, por los mismos motivos que en el punto anterior.

4. Paso : Programa

```
  10 REM    REACCION QUIMICA
  20 BORDER 0: PAPER 0: INK 7: CLS
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA  REPRESE
NTAR  UNA"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"REACCION QUIMICA.
      "
  45 PRINT : PRINT "    -------------------------  "
  47 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  50 PRINT "    -------------------------  "
  60 PRINT AT 9,6; BRIGHT 1; INK 1;"O"
  70 PRINT AT 10,22; BRIGHT 1; INK 1;"O"; INK 5; BRIGHT 1;"O
"
  80 PRINT AT 11,9;"+  "; INK 5; BRIGHT 1;"O"; INK 5; BRIGHT
 1;"O"
  90 PRINT AT 11,17;"="
 100 PRINT AT 12,22; BRIGHT 1; INK 1;"O"; INK 5; BRIGHT 1;"O
"
 110 PRINT AT 13,6; BRIGHT 1; INK 1;"O"
 120 PRINT AT 16,5;"2 C +  O    =   2 CO"
 130 PRINT AT 17,13;"2"
 140 PRINT #0; INK 2; BRIGHT 1;" PULSE UNA TECLA PARA CONTIN
UAR"
 150 IF INKEY$="" THEN  GO TO 150
 160 CLS
```

```
 170 PRINT AT 2,27; INK 5; BRIGHT 1;"O"
 180 PRINT AT 3,20; INK 3; BRIGHT 1;"O";AT 3,26; BRIGHT 1; I
NK 1;"O"
 190 PRINT AT 4,27; INK 5; BRIGHT 1;"O"
 200 PRINT AT 6,5; INK 5; BRIGHT 1;"O";AT 6,11; INK 1; BRIGH
T 1;"O"; INK 5; BRIGHT 1;"O"
 210 PRINT AT 7,4; INK 3; BRIGHT 1;"O"
 220 PRINT AT 7,27; INK 5; BRIGHT 1;"O"
 230 PRINT AT 8,5; INK 5; BRIGHT 1;"O";"  +  "; INK 1; BRIGH
T 1;"O"
 235 PRINT AT 8,15;"="
 240 PRINT AT 8,12; INK 5; BRIGHT 1;"O"
 250 PRINT AT 8,23;"+  "; INK 1; BRIGHT 1;"O"
 260 PRINT AT 9,4; INK 3; BRIGHT 1;"O"
 270 PRINT AT 9,27; INK 5; BRIGHT 1;"O"
 280 PRINT AT 10,5; INK 5; BRIGHT 1;"O"
 290 PRINT AT 10,11; INK 1; BRIGHT 1;"O"; INK 5; BRIGHT 1;"O
"
 300 PRINT AT 12,27; INK 5; BRIGHT 1;"O"
 310 PRINT AT 13,20; INK 3; BRIGHT 1;"O"
 320 PRINT AT 13,26; INK 1; BRIGHT 1;"O"
 330 PRINT AT 14.27; INK 5; BRIGHT 1;"O"
 340 PRINT AT 18,3;"Fe O + 3 CO =  2 Fe + 3 CO"
 350 PRINT AT 19,5;"2 3"
 360 PRINT AT 19,29;"2"
 365 FOR P=0 TO 15: BEEP .06,P: NEXT P
 370 PRINT #0; FLASH 1; PAPER 6; INK 4;"        FIN DEL PROG
RAMA        "
 380 PAUSE 200
```

5. PASO : Lista de variables

No se utiliza ninguna variable.

6. PASO : Descripción del programa

Línea 10-50 : Título

Línea 60 : Impresión de una bolita azul en la línea 9, columna 6.

Línea 70 : Impresión de una bolita azul y de una bolita cyan en las columnas 22/23, línea 10

Línea 80 : Impresión de dos bolitas en las columnas 9 y 10, línea 11

Línea 90 : Impresión de un signo = en la columna 17, línea 11

Línea 100 : Igual que en la línea 70, pero en la línea 12

Línea 110 : Igual que en la línea 60, pero en la línea 13

Línea 120-130 : Impresión del texto:
2 C + O = 2 CO

Línea 140-160 : Espera y borrado de la pantalla

Línea 170-330 : Impresión del gráfico de la segunda reacción

Línea 340-360 : Impresión de la segunda reacción

Línea 365 : Bucle generador de sonido

Línea 370-380 : Final del programa

7. PASO : Resultados

El programa presentado genera consecutivamente dos gráficos en la pantalla; el segundo dibujo aparece cuando el usuario pulsa cualquier tecla.

Podemos prescindir de presentar aquí ambos dibujos, pues no nos es posible efectuar una representación en colores. Nos parece más razonable que el lector pruebe directamente el programa en un televisor o monitor.

3.4 Cálculo estequiométrico

1. PASO : Presentación del problema

Los cálculos estequiométricos consisten en cuantificar las cantidades de las diterentes sustancias que participan en una reacción química. Permiten contestar por ejemplo a la siguiente cuestión:

Cuántos gramos de azufre son necesarios para convertir completamente en sulfuro de zinc 100 gramos de zinc? (Azufre = S; Zinc = Zn; Sulfuro de zinc = ZnS).

Peso atómico del azufre : 32.1
Peso atómico del zinc : 65.4

2. PASO : Análisis del problema

Para resolver el problema deberemos partir de la correspondiente reacción:

$$Zn + S \longrightarrow ZnS$$

Si relacionamos las llamadas masas molares de la substancia buscada y de las subtancias dadas, obtendremos que:

son necesarios 32.1 g de azufre para
transformar 65.4 g de zinc (ver arriba).

Se cumple que:

$$x/100 = 32.1/65.4$$

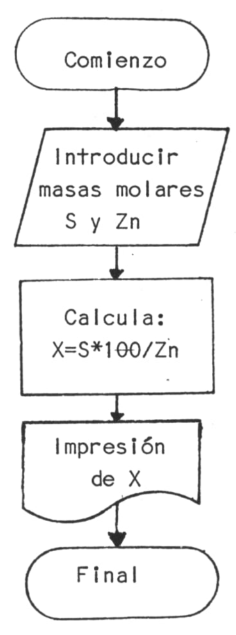
o

$$x = 3210/65.4$$

x es el valor buscado.

3. PASO : Diagrama de flujo

En este problema el diagrama de flujo vuelve a ser muy sencillo:



4. Paso : Programa

```
 10 REM   CALCULOS QUIMICOS
 20 BORDER 0: PAPER 0: INK 7: CLS
 25 REM PONER EL CURSOR EN MODO MAYUSCULAS Y DAR MAS SONIDO
DE LAS TECLAS
 26 POKE 23658,8: POKE 23609,35
 30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA REALIZAR
CALCULOS"
 40 PRINT  PAPER 4; INK 0; FLASH 1;"ESTEQUIOMETRICOS.
      "
 45 PRINT : PRINT "    -----------------------  "
 47 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
 50 PRINT "     ----------------------   "
 60 PRINT AT 7,11; FLASH 1; INK 2; BRIGHT 1;" EJEMPLO "
 70 PRINT : PRINT "ZINC + AZUFRE  = SULFURO DE ZINC"
 80 PRINT : PRINT " Zn  +   S    =      Zn S"
 90 PRINT AT 15,0;"Cuantos gramos de azufre son ne-"
100 PRINT "cesarios para transformar 100 gr"
110 PRINT "de Zinc completamente en Sulfuro"
112 PRINT "de Zinc ?"
120 INPUT "      1. ELEMENTO  :  "; LINE A$
130 INPUT "      2. ELEMENTO  :  "; LINE B$
140 INPUT ("  PESO ATOMICO DEL ";A$;" ? ");A
150 INPUT ("  PESO ATOMICO DEL ";B$;" ? ");B
160 LET X=B*100/A
170 FOR P=0 TO 3: PRINT AT 15+P,0;"
     ": NEXT P
180 PRINT AT 16,3;"CANTIDAD NECESARIA DE ZINC :"
190 PRINT AT 18,10;X;"  gr."
200 FOR P=0 TO 15: BEEP .06,P: NEXT P
210 PRINT #0; FLASH 1; PAPER 6; INK 4;"       FIN DEL PROG
RAMA      "
220 PAUSE 200
```

5. PASO : Lista de variables

A = Peso atómico elemento A

A$ = Nombre del elemento A

B = Peso atómico elemento B

B$ = Nombre del elemento B

X = Cantidad buscada

6. PASO : Descripción del programa

Línea 10-112 : Título y visualización del ejemplo del que se ocupa el programa

Línea 120-130 : Introducción del nombre de ambas sustancias

Línea 140-150 : Impresión de los nombres de las sustancias y demanda de los pesos atómicos de las mismas

Línea 160 : Cálculo de la cantidad buscada

Línea 170 : Bucle de impresión de 4 líneas en blanco a partir de la línea 15

Línea 180-190 : Impresión del resultado

Línea 200 : Bucle generador de sonido

Línea 210-220 : Final del programa

7. PASO : Resultados

Después de las aclaraciones, el programa pide el nombre del primer elemento.

Si introducimos p.e. "azufre", entonces nos pide el nombre del segundo elemento que deberá considerarse en el cálculo.

Si introducimos p.e. "zinc", el programa nos obliga a introducir los pesos atómicos del azufre y del zinc.

Si en respuesta introducimos los valores 32.1 y 65.4, el programa nos ofrece el siguiente resultado:

CANTIDAD NECESARIA DE ZINC :

203.738 gr.

FIN DEL PROGRAMA

Se entiende que el programa debe ser modificado profundamente en determinados casos, cuando se trata de procesos químicos distintos a este (p.e. también cuando participan más de dos substancias en las reacciones del problema).

En este caso hemos procurado únicamente presentar a modo de ejemplo un programa de este tipo.

3.5 El sistema periódico de los elementos

1. PASO : Presentación del problema

No sólo podemos utilizar un ordenador para efectuar cálculos como el del ejemplo precedente o para generar gráficos ilustrativos (como en el punto anterior), sino que puede utilizarse además - el ámbito escolar - precisamente como "fichero de consulta".

El ejemplo del sistema periódico de los elementos ilustra el posible aspecto que cubre un ''programa de consulta'' de este tipo, que también puede entenderse como programa de entreno.

El usuario puede desear, por ejemplo, que el ordenador le dé informaciones básicas acerca de un elemento previamente introducido (p.e. el azufre).

Estas informaciones pueden ser por ejemplo:

1. Símbolo químico,
2. Número de orden en el sistema periódico,
3. Peso atómico,
4. Peso específico
5. Punto de fusión,
6. Punto de evaporación,
7. Grupo (I-VIII,0),
8. Subgrupo (a,b,),
9. Capas de electrones (K-Q)

2. PASO : Análisis del problema

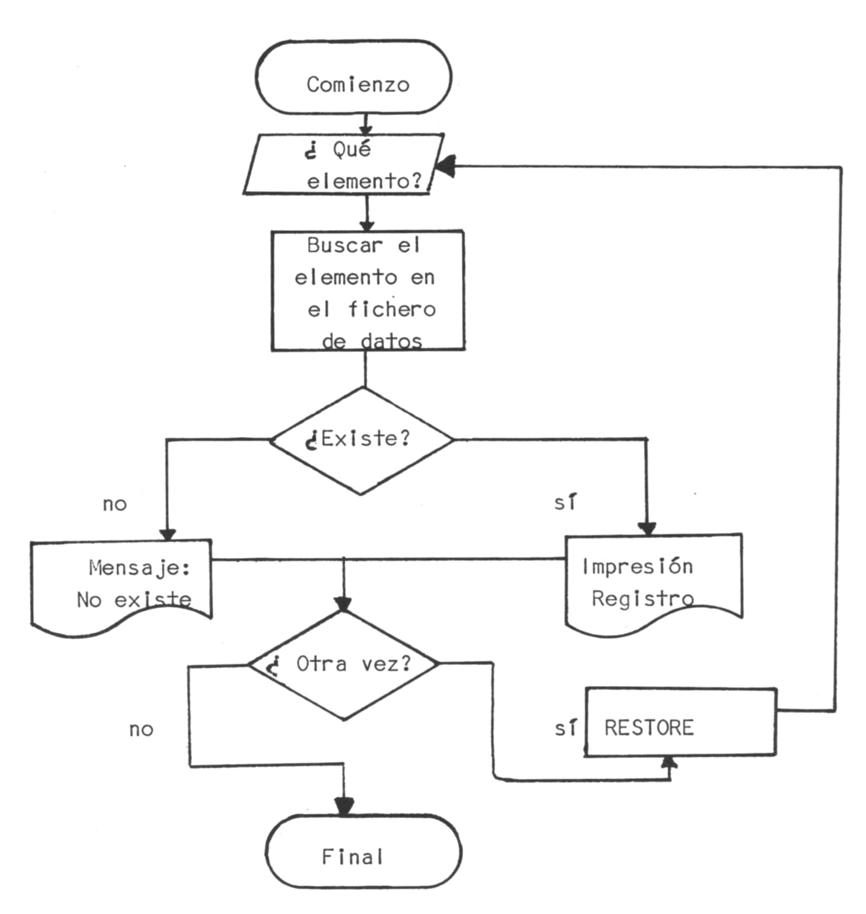
El desarrollo de un programa que nos pueda proporcionar estas informaciones no supone dificultades especiales.

El usuario indicará a través del INPUT el elemento del que desea obtener informaciones y el programa pondrá a su disposición el registro completo asignado a este elemento, extrayéndolo de los DATAS mediante la instrucción READ.

Finalmente puede consultar si se desean obtener las mismas informaciones de otro elemento químico. Si es así, puede accederse nuevamente al fichero mediante RESTORE.

3. PASO : Diagrama de flujo

El diagrama de flujo es igual de sencillo que el análisis del problema efectuado en el párrafo anterior:



4. PASO : Programa

```
 10 REM    Q4-ELEMENTOS
 20 BORDER 0: PAPER 0: INK 7: CLS
 25 REM PONER EL CURSOR EN MODO MAYUSCULAS Y DAR MAS SONIDO
 DE LAS TECLAS
 26 POKE 23658,8: POKE 23609,35
 30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA INDICAR C
ARACTERIS"
 40 PRINT  PAPER 4; INK 0; FLASH 1;"TICAS FUNDAMENTALES DE
LOS  ELE-"
 42 PRINT  PAPER 4; INK 0; FLASH 1;"MENTOS QUIMICOS.
"
 45 PRINT : PRINT "    -----------------------  "
 47 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
 50 PRINT "    -----------------------  "
 60 LET N=16: REM NUMERO DE ELEMENTOS REGISTRADOS
 70 DIM E$(20): DIM C$(2): DIM G$(7): DIM N$(1): DIM S$(1)
 80 DIM W$(7): DIM B$(20): DIM K$(7): DIM L$(4)
 90 INPUT "NOMBRE DEL ELEMENTO QUE SE DESEACONSULTAR  :   "
; LINE B$
100 RESTORE
110 FOR I=1 TO N
120 READ E$: READ C$: READ O$: READ G$: READ W$: READ J$: R
EAD K$: READ L$: READ N$: READ S$
130 IF E$=B$ THEN  GO TO 160
140 NEXT I
150 PRINT AT 13,5; FLASH 1;"ELEMENTO NO REGISTRADO": GO TO
290
160 CLS : PRINT
170 PRINT AT 1,4; FLASH 1; PAPER 2; INK 5;" ";E$;C$;" "
180 PRINT : PRINT : PRINT : PRINT
190 PRINT "-NUMERO ATOMICO          : "; INK 5; BRIGHT 1;O$
200 PRINT "-GRUPO                   : "; INVERSE 1;L$
210 PRINT "-SUBGRUPO                : "; INK 2; BRIGHT 1;N$
220 PRINT "-CAPA                    : "; INK 2;S$
230 PRINT : PRINT
240 PRINT "-PESO ATOMICO            : "; INK 4; BRIGHT 1;G$
250 PRINT "-PESO ESPECIFICO         : "; INK 6; BRIGHT 1;W$
260 PRINT : PRINT
270 PRINT "-PUNTO DE FUSION         : "; INK 3; BRIGHT 1;J$
280 PRINT "-PUNTO DE EVAPORACION    : "; INK 3;K$
290 PRINT #0;"       OTRO ELEMENTO (S/N) "
300 LET A$=INKEY$
310 IF A$="S" THEN  CLS : GO TO 10
320 IF A$="N" THEN  INPUT 0: GO TO 340
330 GO TO 300
340 PRINT #0; FLASH 1; PAPER 6; INK 4;"        FIN DEL PROG
```

```
RAMA          ": PAUSE O
 500 DATA "ALUMINIO","Al","13","26.97","2.7","658","2270","I
II","B","M"
 510 DATA "BARIO","Ba","56","137.36","3.5","704","1537","II"
,"A","P"
 520 DATA "PLOMO","Pb","82","207.21","11.34","327","1690","I
V","B","P"
 530 DATA "CLORO","Cl","17","35.457","1.557","-100","-34","V
II","B","M"
 540 DATA "HIERRO","Fe","26","55.85","7.86","1525","2450","V
III","O","N"
 550 DATA "FLUOR","F","9","19",".0017","-218","-187","VII","
B","L"
 560 DATA "ORO","Au","79","197","19.25","1063","2677","I","B
","P"
 570 DATA "HELIO","He","2","4.003",".000018","-272","-269","
O","O","K"
 580 DATA "IODO","I","53","126.92","4.942","114","184","VII"
,"B","O"
 590 DATA "CADMIO","Cd","48","112.41","8.64","321","770","II
","B","O"
 600 DATA "SODIO","Na","11","22.991",".971","98","880","I","
A","M"
 610 DATA "FOSFORO","P","15","30.98","1.83","44","280","V","
B","M"
 620 DATA "OXIGENO","O","8","16",".0014","-218.7","-182.97",
"VI","B","L"
 630 DATA "AZUFRE","S","16","32.066","2.07","112","444","VI"
,"B","M"
 650 DATA "HIDROGENO","H","1","1.008",".00009","-262","-252.
78","I","A","K"
 660 DATA "ZINC","Zn","30","65.38","7.14","419","906","II","
B","N"
```

Debe recordarse que el fichero de datos de este programa contiene únicamente 16 elementos. No obstante, puede descubrirse con facilidad el modo de variar el programa cuando quieran incorporarse más de 16, o quizá todos los elementos químicos conocidos.

5. PASO : Lista de variables

A$ = Cadena de respuesta (si/no)
AB$ = Abreviación del elemento
B$ = Campo para acoger el elemento buscado por el usuario
E$ = Nombre del elemento
G = Peso atómico
G$ = Grupo a que pertenece el elemento
I = Indice variable
N = Número de elementos del registro
N$ = Subgrupo
O = Número de orden
S1 = Punto de fusión
S2 = Punto de evaporación
W = Peso específico

6. PASO : Descripción del programa

Línea 10-50 : Título

Línea 60 : Indicación del número de elementos (este programa no incluye todos los elementos químicos, sino sólo 16. Dado el caso habrá que modificar la línea 60 si se quieren incorporar más DATAS)

Línea 70-80 : Dimensionado

Línea 90 : Demanda del elemento del que se desea información

Línea 100 : Restauración del registro (para eventualmente repetir el proceso ver línea 290)

Línea 110-140: Lectura de los registros

Línea 130 : Si se ha hallado el elemento, salto hacia 160

Línea 150 : La línea 150 se alcanza únicamente cuando el elemento buscado no se encuentra en el fichero; después se salta a la línea 290

Línea 160-280: Impresión de las informaciones una vez borrada la pantalla

Línea 290-330 : Consulta si se desea consultar otro fichero. En caso afirmativo borrar la pantalla y saltar a la línea 80. En caso negativo seguir en 300.

Línea 340 : Finalización del programa

Líneas 500
y siguientes : Fichero en forma de DATAS

7. PASO : Resultados

Introduzcamos por ejemplo el elemento azufre, a lo cual el ordenador contestará:

```
     AZUFRE          S
---------------------------

Número atómico     =  16
Grupo              =  VI
Subgrupo           =  B
Capa               =  M

Peso atómico       =  32.066
Peso específico    =  2.07

Punto de fusión    =  112
Punto evaporación  =  444


   OTRO ELEMENTO (S/N)
```

Capítulo 4 : Física

4.1 Consideración previa

El campo de la física a nivel escolar, ofrece problemas similares a los que ya conocemos del capítulo dedicado a las matemáticas (ver capítulo 2). En todo caso, aquí deben considerarse ciertas particularidades que nos obligan a hacer uso nuevamente de las posibilidades gráficas del ordenador.

Por esta razón - y ampliando lo dicho en el capítulo anterior con respecto a los métodos de programación de gráficos - en esta introducción profundizaremos otra vez en las instrucciones gráficas del BASIC.

4.2 Programación de gráficos

El SPECTRUM ofrece dos opciones distintas para la creación de gráficos:

1. Gráficos normales (llamados también gráficos de bloque)

2. Gráficos de alta resolución

Para la realización de gráficos en baja resolución, pueden usarse los caracteres gráficos predefinidos en la ROM del ordenador, o bien, caracteres gráficos definidos por el usuario en la RAM o GDU's. También es posible utilizar para la confección de gráficos de bloques, el carácter o caracteres ''espacio'', combinados con distintos colores de fondo.

Ejemplo :

```
10 FOR X=0 TO 21
20 PRINT AT X,10; PAPER 3;"           "
30 NEXT X
```

Este pequeño programa imprime una columna violeta en el centro de la pantalla.

Si dibujamos gráficos de bloque podemos imaginar a la pantalla como una matriz de 24 filas x 32 columnas, es decir, con un total de 768 elementos o bloques de 8 x 8 píxels, numerados de la siguiente forma:

```
(0,0)  (0,1) ..................... (0,31)
(1,0)  (1,1) ..................... (1,31)
  .      .                            .
  .      .                            .
  .      .                            .
  .      .                            .
  .      .                            .
(21,0) (21,1) ................... (21,31)
```

El acceso a las filas 23 y 24 es más problemático pués, coincide con la zona que se reserva el ordenador para imprimir mensajes.

No obstante podemos usar esta zona usando las sentencias PRINT ≠ 0 -para la fila 22-, y PRINT ≠ 1 -para la fila 23-, en combinación con la sentencia auxiliar TAB.

Si se usa la pantalla en la modalidad de alta resolución, el eje de abcisas (X), se divide en 256 píxels, -numerados del 0 al 255-, y el de ordenadas (Y), en 176 píxels ,-numerados del 0 al 175-.

El origen del eje de coordenadas, se sitúa, en alta resolución, en el ángulo inferior izquierdo, a diferencia del de baja resolución, que se situa en el ángulo superior izquierdo.

```
(175,0) (175,1)........................(175,255)
   .                                       .
   .                                       .
 (0,0)  (0,1)  ........................(0,255)
```

También se puede dibujar en alta resolución, en la zona reservada a la impresión de mensajes del ordenador. Esto es posible desde el BASIC, ''pokeando'', en las direcciones del archivo de pantalla comprendidas entre el 22016 y el 22527, inclusives, y en las zonas del fichero de atributos comprendidas entre las direcciones 23232 y 23295, incluidas ambas.

Para dibujar en alta resolución, el SPECTRUM dispone de las sentencias PLOT, DRAW y CIRCLE, que pueden ser combinadas con los comandos INK, PAPER, FLASH, BRIGHT, OVER e INVERSE.

En el SPECTRUM pueden seleccionarse 8 colores distintos, en dos gamas diferentes de tonos, lo que nos proporciona un total de 15 colores posibles (el negro no tiene dos tonos).

Estos colores se asignan mediante el uso de los comandos PAPER e INK, que como su traducción indica, se refieren al color del papel o fondo de la pantalla, y al de la tinta de la misma.

El formato de ambos comandos es:

PAPER código

INK código

Los códigos de los distintos colores que se pueden seleccionar son los siguientes:

Código	Color	Código	Color
0	Negro	4	Verde
1	Azul claro	5	Azul oscuro
2	Rojo	6	Amarillo
3	Magenta	7	Blanco

El color del borde de la pantalla puede seleccionarse mediante el comando: BORDER código.

No es posible dar un color a cada píxel de forma independiente, solo pueden asignarse dos colores (PAPER e INK) por cada elemento de la pantalla en baja resolución.

Instrucción 18:
----

nn PLOT (coordenada X,coordenada Y)

Marca un pixel en las coordenadas especificadas

Instrucción 19:
----

nn DRAW (coordenada X,coordenada Y),(arco)

Dibuja una línea recta, o un arco si se incluye el tercer parámetro. El punto de origen es el último píxel procesado, por una de las sentencias PLOT DRAW o CIRCLE, y el punto de destino es el especificado por las coordenadas, -relativas al punto de origen, (en torno al cual podemos situar un nuevo origen de coordenadas)-, del argumento de DRAW. Las sentencias CLS, NEW, CLEAR y RUN posicionan de nuevo el origen en el punto 0,0.

Instrucción 20:
----

nn CIRCLE (coordenada X,coordenada Y), (R)

Dibuja un círculo de centro X,Y y radio = R

Instrucción 21

OVER (código de control)

Esta sentencia controla la sobreimpresión de los caracteres.
Cuando el código de control es 'cero', se borra lo que anteriormente hubiera en esa posición. Si el código de control es uno, se combinan uno a uno los píxels del caracter anterior y los del nuevo. Esto se consigue mediante el uso de la función lógica XOR.

4.3 Programas BASIC

```
10 REM
20 REM          CUADRADO
30 REM
40 BORDER 0: INK 7: PAPER 0: CLS
50 PLOT 80,40
60 DRAW 100,0
70 DRAW 0,100
80 DRAW -100,0
90 DRAW 0,-100
95 PAUSE 0
96 CLS
```

```
100 REM
110 REM    RECTAS AL AZAR
120 REM
125 PRINT #0;TAB 3;"PULSE UNA TECLA PARA
130 LET X=INT (RND*256-128)
140 LET Y=INT (RND*176-96)
145 LET C=RND*7
150 PLOT 128,96
160 DRAW  INK C; OVER 1;X,Y
165 IF INKEY$<>"" THEN  GO TO 180
170 GO TO 130
180 PAUSE 0
185 CLS
190 REM
200 REM   EJES DE COORDENADAS
210 REM
220 FOR X=0 TO 255 STEP 5
230 PLOT X,87
240 NEXT X
250 FOR Y=0 TO 175 STEP 5
260 PLOT 128,Y
270 NEXT Y
280 PAUSE 0
```

Este programa consta de tres subprogramas encadenados. El primero dibuja un cuadrado de lado = 100 píxels. El segundo traza rectas al azar, cuyo origen se sitúa en el punto (128,96), asimismo, se selecciona también al azar el color en que se traza la recta. El tercero dibuja mediante dos bucles FOR-NEXT unos ejes coordenados.

Este programa dibuja píxel a píxel y en diferentes colores un cuadrado. Fíjese que aunque se intenta dibujar cada píxel en un color diferente, esto no es posible en el modo de gran resolución, lo único que se logra es colorear diferenciadamente -a lo máximo con dos colores-, bloques horizontales de 8 x 8 píxels.

```
10 BORDER 0: INK 7: PAPER 0: CLS
20 FOR S=80 TO 173
30 FOR Z=47 TO 155
40 LET F=INT (RND*7)
50 PLOT  INK F;S,Z
60 NEXT Z
70 NEXT S
80 PAUSE 0
```

El siguiente programa traza una trama reticular en la pantalla del televisor:

```
 10 BORDER 0: INK 7: PAPER 0: CLS
 15 FOR C=1 TO 7
 20 FOR X=0 TO 255 STEP 5
 30 PLOT  INK C;X,0
 40 DRAW  INK C;0,175
 50 NEXT X
 60 FOR Y=0 TO 175 STEP 5
 70 PLOT  INK C;0,Y
 80 DRAW  INK C;255,0
 90 NEXT Y
100 NEXT C
110 PAUSE 0
```

El próximo programa traza aleatoriamente una trama de cuadrados de distintos tamaños.

```
 10 BORDER 0: INK 7: PAPER 0: CLS
 15 FOR X=1 TO 200
 20 LET A1=INT ((207-0+1)*RND)+0
 30 LET A2=INT ((127-0+1)*RND)+0
 40 LET LONGITUD=INT (RND*31)
 50 PLOT A1,A2
 60 DRAW LONGITUD,0
 70 DRAW 0,LONGITUD
 80 DRAW -LONGITUD,0
 90 DRAW 0,-LONGITUD
100 NEXT X
110 PAUSE 0
```

El siguiente programa traza tres columnas horizontales de distintos colores:

```
 10 BORDER 0: INK 7: PAPER 0: CLS
 15 BRIGHT 1
 20 FOR X=0 TO 21
 30 PRINT  PAPER 1;"    "; PAPER 2;"    "; PAPER 3;"    ";
PAPER 4;"    "; PAPER 5;"    "; PAPER 6;"    "; PAPER 7;"
  "
 40 NEXT X
 50 PRINT #0;TAB 8;"FIN DEL PROGRAMA"
 60 PAUSE 0
```

El siguiente programa dibuja un rectángulo y sus diagonales:

```
10 BRIGHT 1: BORDER 0: INK 7: PAPER 0: CLS
20 FOR A=0 TO 200
30 LET X=INT ((207-47+1)*RND)+47
40 LET Y=INT ((127-47+1)*RND)+47
50 LET R=INT (RND*31)
55 LET C=INT (RND*8)
60 CIRCLE  INK C;X,Y,R
70 NEXT A
90 PAUSE 0
```

Este otro sirve para ilustrar el uso de la sentencia CIRCLE, y produce una serie de circunferencias, de posición y tamaño aleatorio, y conformando en pantalla, un bonito efecto artístico.

```
10 BORDER 0: INK 7: PAPER 0: CLS
20 DRAW 255,0
30 DRAW 0,175
40 DRAW -255,0
50 DRAW 0,-175
60 DRAW 255,175
70 PLOT 255,0
80 DRAW -255,175
90 PAUSE 0
```

El programa siguiente imprime en pantalla y con distintos colores, 1000 puntos. Se consigue de este modo un efecto semejante al que producen las estrellas en la esfera celeste.

```
10 BRIGHT 1: BORDER 0: INK 7: PAPER 0: CLS
20 FOR I=1 TO 1000
30 LET X=INT (RND*256)
40 LET Y=INT (RND*176)
50 LET C=INT (RND*8)
60 PLOT  OVER 1; INK C;X,Y
70 NEXT I
80 PAUSE 0
```

Este programa dibuja polígonos de cualquier número de lados.

```
 10 INPUT TAB 7;"NUMERO DE LADOS = ";N
 15 IF N<3 THEN  CLS : GO TO 10
 20 INPUT TAB 10;"LONGITUD = ";L
 25 IF L>175 THEN  GO TO 20
 30 PLOT 126,0
 40 FOR X=1 TO N
 50 DRAW L*COS (((X-1)/N)*2*PI),L*SIN (((X-1)/N)*2*PI)
 60 NEXT X
 70 PRINT AT 0,10; FLASH 1; PAPER 2; INK 4;"OTRA VEZ S/N"
 80 LET A$=INKEY$
 90 IF A$="S" OR A$="s" THEN  CLS : GO TO 10
100 IF A$="N" OR A$="n" THEN  CLS : GO TO 120
110 GO TO 80
120 PRINT AT 11,8; FLASH 1; PAPER 5; INK 2;"FIN DEL PROGRAM
A"
```

Habrá observado sin duda alguna, que en muchos de los programas que ilustran este capítuló, aparece la instrucción PAUSE 0.

Esta instrucción tiene como misión inducir al ordenador a permanecer en estado de espera, o mejor dicho, en un bucle sin salida, mientras que no se aprete una tecla cualquiera. Este comando pues, espera a que se introduzca un caracter a través del teclado y mientras esto no ocurre, retiene la ejecución del programa en el ordenador.

La instrucción PAUSE retiene el desarrollo de un programa según el número fijado en su argumento. Así, PAUSE 0, retiene de modo indefinido un programa. Para módulos > 0 cada fracción de 50 equivale a un segundo de espera. De este modo si establecemos PAUSE 50 el programa se detendrá durante un segundo, si hacemos PAUSE 100 el programa se parará durante dos segundos, etc......

Si no induciésemos de algún modo al ordenador a entrar en estado de espera o en un bucle sin salida, no lograríamos llegar a visualizar nada en la ventanilla destinada a la impresión de los mensajes del sistema -líneas 22 y 23 de la pantalla-, ya que el ordenador imprimiría después de finalizado el programa el consiguiente OK., lo que motivaría el borrado de esta zona, sin que tengamos tiempo de visualizarla.

Se podría conseguir también el mismo efecto incluyendo en el programa una línea del tipo:

```
nn   GOTO nn
```

Lo que haría que el ordenador cayese en un bucle sin fin. Para salir del mismo será necesario provocar un BREAK mediante el uso de las teclas CAPS SHIFT + SPACE.

4.4 El dinamómetro

1. PASO : Presentación del problema

Si colgamos diferentes objetos de un dinamómetro, éste se estira en mayor o menor grado, según sea el peso de los cuerpos.

Así se cumple una ley de proporcionalidad que dice: si doblamos el peso, el alargamiento del dinamómetro se duplica.

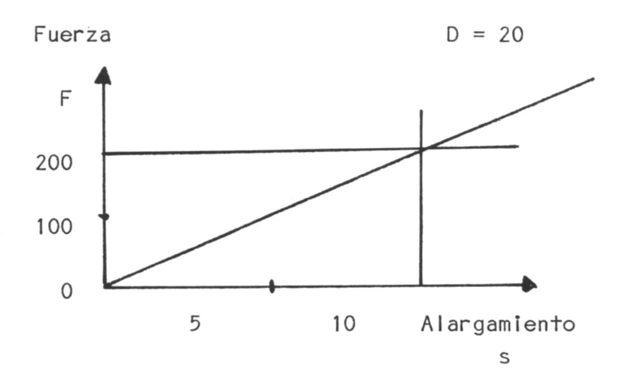
Existe también una llamada constante elástica, que es igual al cociente entre el peso F y la deformacion del muelle s, es decir:

$$D = F/s = \text{constante}$$

Si, por ejemplo, D = 20, una fuerza de 100 (cN) producirá un alargamiento de 5 cm, y una fuerza de 200 (cN) un alargamiento del muelle de 10 cm etc.

2. PASO : Análisis del problema

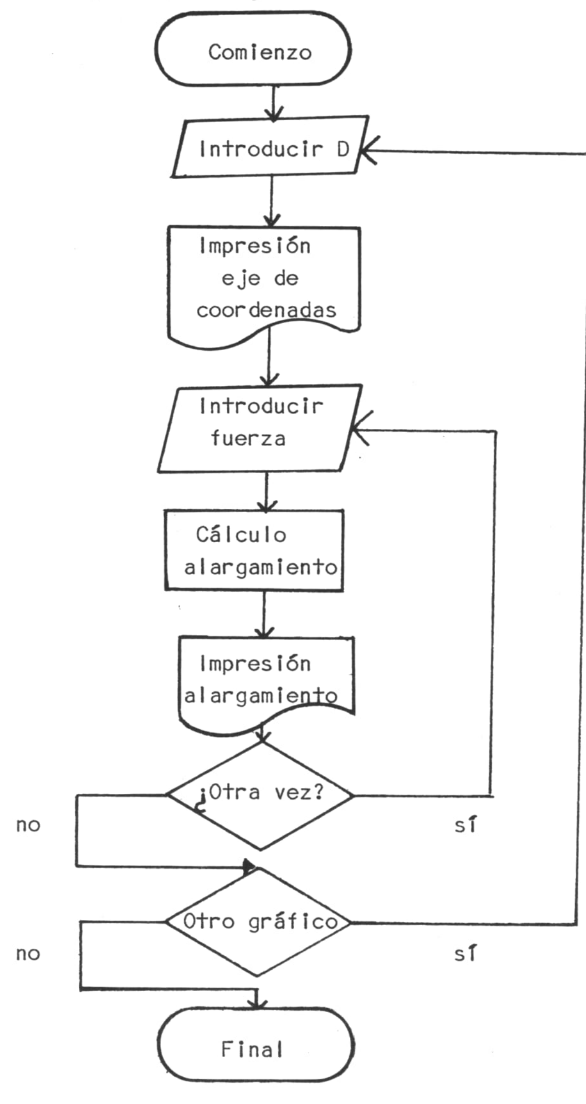
La relación antes descrita puede ser representada gráficamente (p.e. para D = 20):



El programa BASIC a desarrollar, debe imprimir el gráfico correspondiente a diversos valores de la constante de muelle D en la pantalla, siendo necesario estructurarlo del modo siguiente:

- Trazado de unos ejes de coordenadas,
- introducción de un valor arbitrario para D,
- impresión de una recta de constancia (ver arriba),
- contestación a la pregunta referente al alargamiento del dinamómetro tras introducir un peso cualquiera.

3. PASO : Diagrama de flujo



4. Paso : Programa

```
 10 REM    F1-DINAMOMETRO
 20 BORDER 0: PAPER 0: INK 7: CLS
 25 POKE 23658,8: POKE 23609,35
 30 PRINT  PAPER 4; INK 0; FLASH 1;"*PROGRAMA PARA ILUSTRAR
GRAFICA-"
 40 PRINT  PAPER 4; INK 0; FLASH 1;"MENTE LA LEY DE HOOKE.
      "
 45 PRINT : PRINT "    -----------------------  "
 47 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
 50 PRINT "    -----------------------  "
 60 INPUT "VALOR DE LA CONSTANTE ELASTICA ?             ";D
 70 PLOT 19,9: DRAW 215,0
 80 PLOT 19,9: DRAW 0,100
 90 PLOT 19,9
100 LET B=D/10
110 FOR A=19 TO 215
120 LET E=9+B*(A-19)
130 IF E<=9 OR E>110 THEN  GO TO 150
140 PLOT  INK 4; BRIGHT 1;A,E
150 NEXT A
160 INPUT TAB 7;"FUERZA F   = ? ";F
170 LET FW=F/D: LET FW=INT (FW*10000+.5)/10000
180 CLS
190 FOR X=0 TO 20: BEEP .05,X: NEXT X
200 PRINT AT 8,3; INK 5;"CONSTANTE ELASTICA  : ";D
210 PRINT AT 13,3; INK 5;"RECORRIDO           : ";FW
220 PRINT #0;TAB 6; FLASH 1; INK 4;"OTRO CALCULO (S/N)"
230 LET A$=INKEY$
240 IF A$="S" THEN  CLS : GO TO 160
250 IF A$="N" THEN  INPUT 0: GO TO 270
260 GO TO 230
270 PRINT #0;TAB 8; FLASH 1; PAPER 6; INK 4;"FIN DEL PROGRA
MA": PAUSE 0
```

5. PASO : Lista de variables

A = Columna para el trazado de la recta

A$ = Cadena de respuesta (sí/no)

B = D/10 (Magnitud para normalizar la impresión en pantalla)

D = Constante de muelle

E = Línea para el trazado de la línea de constancia

F = Fuerza

FW = Recorrido del muelle

I = Indice variable

6. PASO : Descripción del programa

Línea 10-50 : Título

Línea 60 : Introducción de la constante elástica D

Línea 70-90 : Trazado de los ejes de coordenadas Origen: 19,9

Línea 110-150: Trazado de la recta
No se dibujan los puntos de la recta que están situados fuera de la superficie encerrando los ejes de coordenadas.

Línea 160 : Demanda de la fuerza F

Línea 170 : Cálculo del recorrido del muelle y reducción al formato de 4 decimales.

Línea 180 : Borrado de la pantalla

Línea 190 : Bucle generador de sonido. La variable X controla el tono del mismo.

Línea 200-210: Impresión de los resultados

Línea 220 : Consulta si se desea un nuevo cálculo;

Línea 230 : Se le asigna a la variable alfanumérica A$ el valor de la tecla que se pulse.

Línea 240-250: Se compara la tecla que se ha pulsado:

En caso de pulsarse la ''S'',vuelta a la línea 160 tras borrar la pantalla.

En caso negativo, seguir en línea 270

Línea 260 : Vuelta a la línea 230 en caso de no pulsarse la ''S'' o la ''N''.

Línea 270 : Final del programa

7. PASO : Resultados

Después de introducir la información INPUT, el programa genera un gráfico tal como el que ya presentamos esquemáticamente en una de las páginas precedentes.

Después de una pausa, el programa pide la fuerza F y calcula el correspondiente recorrido de muelle. Sobra aquí cualquier ejemplo de cálculo (comparar con el 1. PASO, que contiene ya un ejemplo).

4.5 El principio de Arquímedes

1. PASO : Presentación del problema

El principio de Arquímedes dice lo siguiente:

El empuje que recibe un cuerpo sumergido en un fluído es igual al peso del líquido que desplaza.

Este principio describe el hecho conocido de que la fuerza necesaria para apretar hacia abajo un cubo vacío dentro de un líquido aumenta progresivamente.

A medida que un cuerpo se sumerge en un líquido, éste se ve sometido a una pérdida de peso aparente cada vez mayor, debida al creciente empuje.

Presentamos a continuación un programa que, basándose en este principio, calcula el volumen y la densidad (peso específico) de un cuerpo desconocido sumergido en un líquido conocido (p.e. agua), una vez se conoce su peso en el aire.

2. PASO : Análisis del problema

Como en todos los demás problemas, aquí se procederá de acuerdo con el probado esquema:

E = Entrada

P = Proceso

S = Salida

Es decir, en primer lugar estudiaremos qué informaciones de entrada necesita el ordenador si debe resolver el problema antes formulado. Necesita lo siguiente:

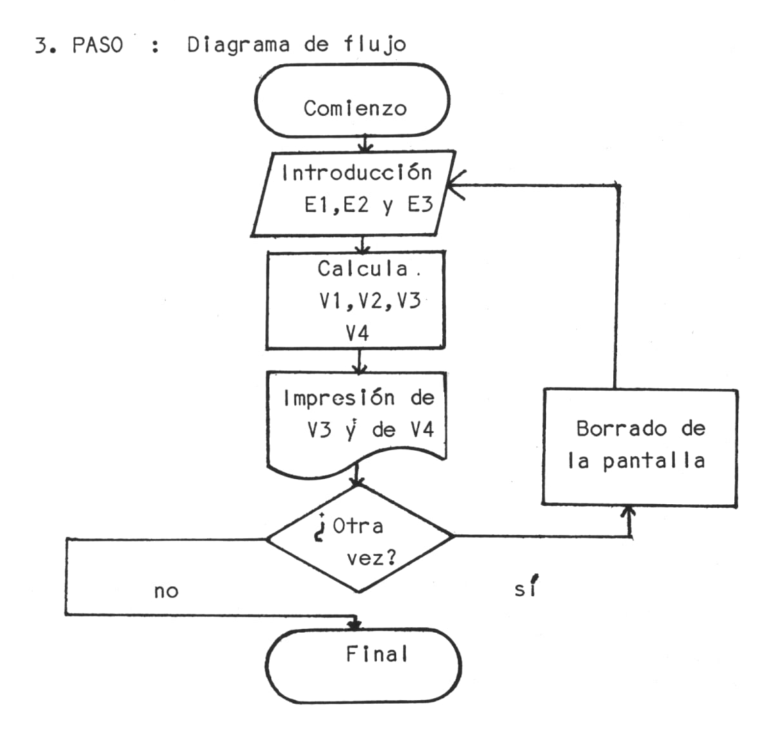
E1. El peso del cuerpo desconocido en el aire;
E2. el peso aparente del cuerpo dentro del fluído, en el que se sumerge;
E3. el peso específico del líquido utilizado para la comprobación.

Cuando ya disponemos de estas informaciones, puede determinarse mediante el proceso lo siguiente:

V1. El empuje y el peso de la cantidad de líquido desplazada;
V2. el volumen de esta cantidad de líquido;
V3. el volumen del cuerpo desplazante de prueba;
V4. El peso específico del cuerpo de prueba sumergido.

Finalmente deberán imprimirse como resultados los obtenidos en los últimos puntos V3 y V4.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
  10 REM    F2-ARQUIMEDES
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"  **PRINCIPIO  DE ARQUI
MEDES**  "
  40 PRINT : PRINT "    ------------------------  "
  45 PRINT AT 3,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  50 PRINT "     ------------------------  "
  60 INK 5: BRIGHT 1: PRINT AT 7,0;"Con el principio  de Arq
uimedes,": PRINT
  70 PRINT "pueden determinarse el volumen y": PRINT
  80 PRINT "la densidad de un cuerpo descono": PRINT
  90 PRINT "cido.": PRINT
 100 INK 7: FLASH 1: PRINT : PRINT AT 16,3;"INTRODUZCA LAS S
IGUIENTES": PRINT
 110 PRINT AT 18,9;"INFORMACIONES": FLASH 0
 120 INPUT "-PESO DEL CUERPO EN EL AIRE ?              ";PA
 130 INPUT "-PESO APARENTE DEL CUERPO EN EL  FLUIDO ? ";PF
 140 INPUT "-PESO ESPECIFICO DEL FLUIDO ?  ";DF
 150 LET PP=PA-PF
 160 LET VD=PP/DF
 170 LET VC=VD
 180 LET DC=PA/VC
 190 CLS
 200 PRINT AT 1,3;"                          "
 210 PRINT AT 2,3;"  "; FLASH 1; PAPER 4; INK 2; BRIGHT 1;"R
ESULTADOS DEL ESTUDIO:"; INK 7; PAPER 0; FLASH 0;"  "
 220 PRINT AT 3,3;"                          "
 230 PRINT AT 9,5;"VOLUMEN       : ";VC
 240 PRINT AT 13,5;"DENSIDAD      : ";DC
 250 PRINT AT 18,6; INK 6;"OTRA VEZ  (S/N) ?"
 260 LET A$=INKEY$
 270 IF A$="S" THEN  CLS : GO TO 10
 280 IF A$="N" THEN  INPUT 0: GO TO 300
 290 GO TO 260
 300 PRINT #0; FLASH 1; PAPER 6; INK 4;"       FIN DEL PROG
RAMA        ": PAUSE 0
```

5. PASO : Lista de variables

A$ = Variable de cadena para las respuestas sí/no
PA = Peso del cuerpo en el aire
PF = Peso aparente del cuerpo en el fluído
PP = Pérdida aparente de peso
VC = Volumen del cuerpo
VD = Volumen del líquido desplazado
DF = Peso específico del líquido
DC = Peso específico del cuerpo

6. PASO : Descripción del programa

Línea 10-110 : Título y anotaciones aclaratorias

Línea 120-140 : Demanda de las informaciones INPUT

Línea 150-180 : Cálculos del resultado

Línea 190-240 : Impresión de los resultados

Línea 250-290 : Consulta si se quiere repetir el problema
Si la respuesta es positiva se borra la pantalla y se va a 10, si no va a la 300 (FIN)

Línea 300 : Final del programa

7. PASO : Resultados

Si introducimos por ejemplo las siguientes informaciones INPUT:

PA: Peso del cuerpo = 70 gr

PF: Peso aparente del cuerpo en el fluído de prueba (a medir con un dinamómetro) = 40 gr

DF: Peso específico del fluido de prueba (agua) =1

El programa genera estos resultados:

```
RESULTADOS DEL ESTUDIO :

Volumen   = 30

Densidad  = 2.333333

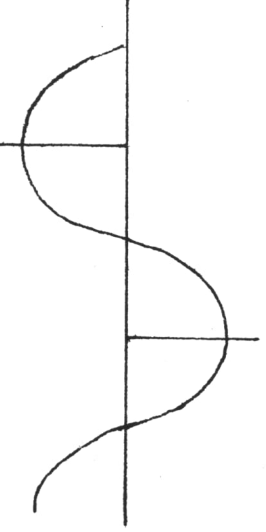
  OTRA VEZ (S/N) ?
```

4.6 Movimiento pendular

1. PASO : Presentación del problema

El movimiento pendular es un fenómeno físico básico que juega un papel importante en la acústica, la óptica y en la electricidad.

Los movimientos pendulares siguen una oscilación sinusoidal, como se muestra en el esquema contiguo.



Presentamos ahora un programa, que representa gráficamente una oscilación sinusoidal para diversas amplitudes (A) y periodos (P), que a su vez determinan la llamada frecuencia (F) de cualquier oscilación.

2. PASO : Análisis del problema

Para el programa previsto deben introducirse la amplitud y el periodo de la oscilación como informaciones INPUT.

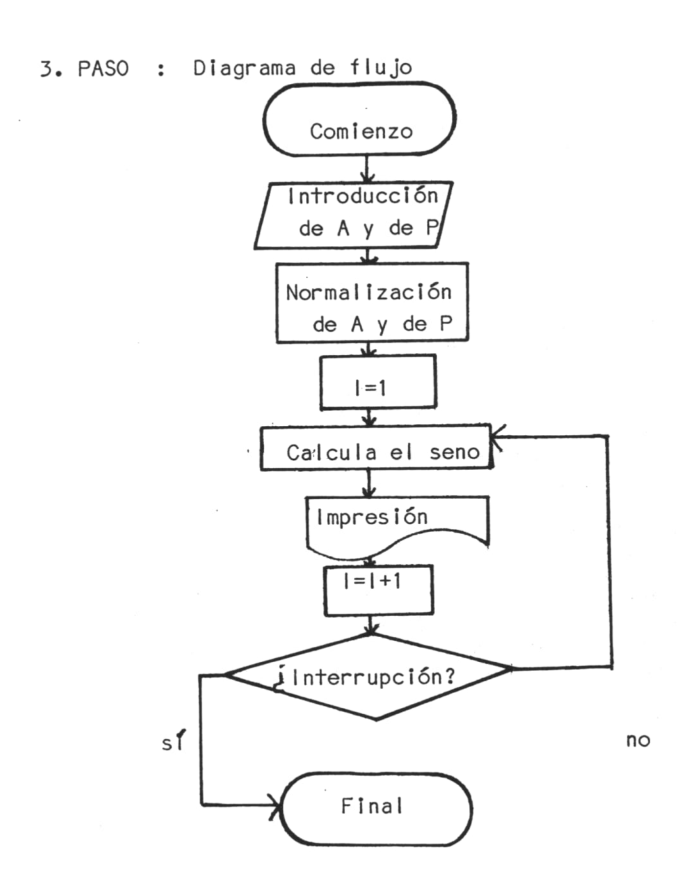
El programa puede ocuparse entonces de que la oscilación sinusoidal "pase", recorriendo la pantalla de arriba a abajo como en el esquema anterior (cosa que resulta más sencilla de programar que si imprimimos la oscilación horizontalmente).

Fundamentalmente, el contenido del programa debe centrarse en la normalización de la amplitud y del período para aprovechar óptimamente la representación gráfica de la oscilación sobre la pantalla.

Esto se consigue colocando la línea central de la oscilación en la columna 16 y ajustando la amplitud, es decir la mayor separación de esta línea, p.e. al valor máximo de 16.

Como puede deducirse de la distribución que acabamos de realizar, no seleccionamos los gráficos de alta resolución (ver Cap. 9), sino que representamos la oscilación con ayuda del caracter gráfico a, definido con la ayuda del programa de la página 97.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
  10 REM    F3-PENDULO
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"*PROGRAMA  PARA LA SIMU
LACION DE"
  35 PRINT  PAPER 4; INK 0; FLASH 1;"UN MOVIMIENTO PENDULAR.
      "
  40 PRINT : PRINT "    ------------------------   "
  45 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  50 PRINT "    ------------------------   "
  54 PRINT AT 13,2;"PARA INTERRUMPIR HACER BREAK"
  60 INPUT "AMPLITUD DE LA OSCILACION ? ";A
  70 INPUT "PERIODO DE LA OSCILACION  ? ";P
  75 IF A>=16 THEN  LET A=16
  80 CLS
  90 LET I=1
 100 LET J=(2*PI/P)*I
 110 LET Y=SIN (J)
 120 LET S=16+Y*A
 130 FOR K=1 TO S
 140 PRINT " ";: POKE 23692,255
 145 REM Este poke inhibe la presentacion del mensaje scroll
? cada vez que se rebasan las 23 lineas de la pantalla
 150 NEXT K
 160 PRINT
 170 LET I=I+1: GO TO 100
```

6. PASO : Descripción del programa

Línea 10-54 : Título y aclaraciones

Línea 60-70 : Demanda de informaciones INPUT

Línea 75 : Limitación de la amplitud de la oscilación al valor máximo de 16

Línea 80 : Borrado de la pantalla

Línea 90 : Comienzo del bucle de impresión (I = índice de contaje de líneas)

Línea 100 : Normalización de I después de tomar el periodo P con la magnitud auxiliar J

Línea 110 : Determinación del correspondiente valor seno

Línea 120 : Determinación de la posición de línea de cada valor de la función en la pantalla, teniendo en cuenta que el valor cero debe estar en la columna 16

Línea 130-150 : Bucle para rellenar la línea con "bolitas" hasta el correspondiente valor de la función

Línea 160-170 : Se fuerza un cambio de línea y se imprime la siguiente línea saltando de nuevo a 100 y elevando I en 1

7. PASO : Resultados

Los resultados que ofrece este programa son análogos al croquis del 1. PASO, con la diferencia de que el espacio entre el margen izquierdo de la pantalla y la sinusoide está lleno de "bolitas" claras.

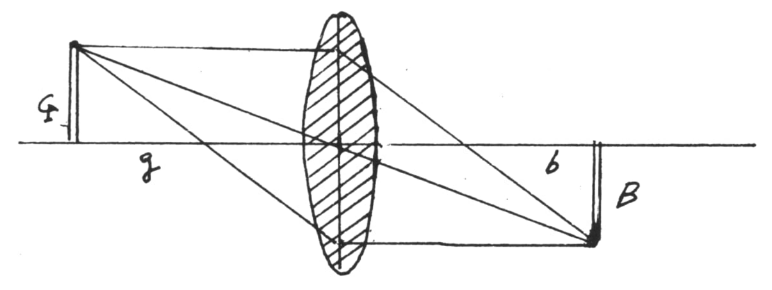
El lector puede descubrir por sí mismo cómo hay que modificar el programa para suprimir este "rellenado", es decir, cómo hay que hacer para marcar ópticamente con "bolitas" sólo aquellas posiciones que realmente se encuentran sobre la sinusoide.

El programa aquí presentado es un programa sin fín, que sólo puede interrumpirse haciendo BREAK. Esta característica también puede modificarse fácilmente.

4.7 Reproducción óptica

1. PASO : Presentación del problema

Una lente convexa nos permite obtener la imagen de cualquier objeto - tal como se muestra en la ilustración siguiente:



La cámara fotográfica funciona según este principio, así como también el ojo humano.

Presentamos a continuación un programa que calcula el tamaño de la imagen y la escala de reproducción de una lente convexa para un tamaño del objeto, una distancia del mismo y una distancia focal de la lente dados.

2. PASO : Análisis del problema

El programa a realizar necesita las siguientes informaciones INPUT:

1. Tamaño del objeto,
2. distancia del objeto,
3. distancia focal de la lente.

Para resolver el problema necesitamos la llamada ecuación óptica:

$$1/f = 1/g + 1/b$$

donde: f = distancia focal
g = distancia del objeto
b = distancia de la imagen

Despejando b en esta ecuación se obtiene:

$$b = \frac{1}{1/f-1/g}$$

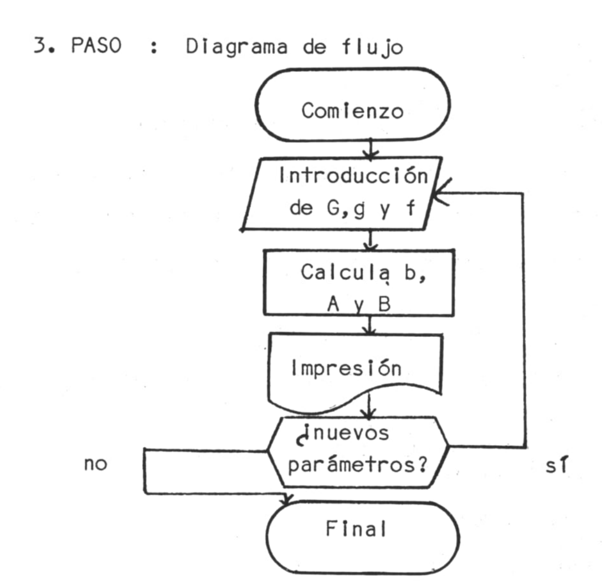
Se cumple también que la escala de reproducción A es igual a:

$$A = B/G = b/g$$

Si el objeto a reproducir tiene por ejemplo un tamaño G = 20 cm, puede determinarse también el tamaño de la imagen con ayuda de las igualdades citadas:

$$B = (b*G)/g, \text{ porque } B/G = b/g$$

3. PASO : Diagrama de flujo



```
  10 REM    F4-OPTICA
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"*PROGRAMA  PARA CALCULA
R LA DIS-"
  35 PRINT  PAPER 4; INK 0; FLASH 1;"TANCIA, EL TAMA/O Y LA
ESCALA DE"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"REPRODUCCION DE LA IMAG
EN PRODU-"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"CIDA POR UNA LENTE CONV
EXA.     "
  50 PRINT : PRINT "     -----------------------     "
  55 PRINT AT 6,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "     ------------------------  "
  70 PRINT AT 13,0; PAPER 3; BRIGHT 1;"INTRODUCE LOS SIGUIEN
TES DATOS :"
  80 INPUT "  -TAMA/O DEL OBJETO    ?  ";G
  90 INPUT "  -DISTANCIA AL OBJETO ?  ";GW
 100 INPUT "  -DISTANCIA FOCAL      ?  ";F
 110 CLS
 120 LET BW=1/(1/F-1/GW)
 130 LET A=BW/GW
 140 LET B=(G*BW)/GW
 145 LET BW=INT (BW*100+.5)/100
 146 LET A=INT (A*100+.5)/100
 147 LET B=INT (B*100+.5)/100
 150 PRINT AT 0,8; PAPER 2; FLASH 1;"DATOS DE PARTIDA"
 160 PRINT AT 3,0;"-Distancia focal          = ";F
 170 PRINT : PRINT "-Tama/o del objeto       = ";G
 180 PRINT : PRINT "-Distancia al objeto     = ";GW
 190 PRINT AT 11,11; PAPER 2; FLASH 1;"RESULTADOS"
 200 PRINT : PRINT : PRINT "-Escala de reproduccion = ";A
 210 PRINT : PRINT "-Tama/o de la imagen     = ";B
 220 PRINT : PRINT "-Distancia a la imagen  = ";BW
 230 PRINT #1;"        OTRO CALCULO (S/N)"
 240 LET A$=INKEY$
 250 IF A$="" THEN  GO TO 240
 260 IF A$="S" THEN  GO TO 10
 270 IF A$="N" THEN  GO TO 280
 275 GO TO 240
 280 INPUT 0: PRINT #0;"        "; FLASH 1; PAPER 2; INK 5;"F
IN DEL PROGRAMA": PAUSE 0
```

6. PASO : Descripción del programa

Línea 10-70 : Título y aclaraciones

Línea 80-100 : Demanda de las informaciones INPUT

Línea 110-140 : Borrado de la pantalla y cálculos

Línea 145-147 : Redondear a dos posiciones decimales

Línea 150-220 : Impresión de los resultados

Línea 230-240 : Consulta si se desean repetir los cálculos y espera una respuesta des de el teclado.

Línea 250-275 : Identifica la entrada que se ha pro ducido desde el teclado y en su caso finaliza el programa.

7. PASO : Resultados

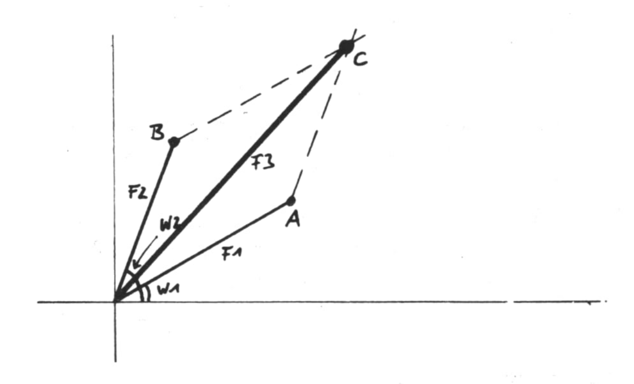
Si introducimos p.e. : G = 20 cm, GW = 300 cm, F = 15 cm, obtendremos:

```
              DATOS DE PARTIDA :
    -Distancia focal                = 15
    -Tamaño del objeto              = 20
    -Distancia al objeto            = 300
              RESULTADOS :
    -Escala de reproducción         = .05
    -Tamaño de la imagen            = 1.05
    -Distancia a la imagen          = 15.79
             OTRO CALCULO (S/N)?
```

4.8 Paralelogramo de fuerzas

1. PASO : Presentación del problema

Cuando se aplican dos fuerzas sobre un punto común, se obtiene una fuerza reultante, que es la diagonal del llamado paralelogramo de fuerzas, como muestra el esquema siguiente:



En el programa que sigue se efectúa la representación gráfica del paralelogramo de fuerzas para cualquier constelación posible de dos fuerzas y se calcula la fuerza resultante F3.

2. PASO : Análisis del problema

El punto de aplicación de ambas fuerzas debe emplazarse en el centro del sistema de coordenadas definido (128,88).

Se introducen las fuerzas F1 y F2, que en primer lugar se optimizarán de modo que el espacio de la pantalla resulte suficiente incluso en una situación poco favorable.

Deben introducirse además los ángulos W1 y W2 con los que se aplican las fuerzas F1 y F2 en el punto de aplicación.

Con ayuda del teorema de Pitágoras y de las relaciones entre ángulos del triángulo rectángulo pueden calcularse los extremos de las componentes vectoriales de las dos fuerzas .

Si se conoce la posición de C puede calcularse asimismo la fuerza R resultante de F1 y F2 (teorema del coseno).

3. PASO : Diagrama de flujo

Aquí puede prescindirse del diagrama de flujo.

```
 10 REM    F-5 PARALELOGRAMO
 20 BORDER 0: PAPER 0: INK 7: CLS
 25 POKE 23658,8: POKE 23609,35
 30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA DIBUJAR U
N PARALE-"
 40 PRINT  PAPER 4; INK 0; FLASH 1;"LOGRAMO DE FUERZAS PARA
 DOS FUER"
 45 PRINT  PAPER 4; INK 0; FLASH 1;"ERZAS APLICADAS.
       "
 50 PRINT : PRINT "    ------------------------   "
 55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
 60 PRINT "    ------------------------   "
 80 PRINT AT 10,1;"-Primera fuerza F1        = ?"
 90 INPUT "            F1 = ";F1
 95 PRINT AT 10,28;F1
100 PRINT AT 12,1;"-Angulo de aplicacion F1 = ?
(0-360)"
105 INPUT "            W1 = ";W1
107 PRINT AT 12,28;W1
110 PRINT AT 17,1;"-Segunda fuerza F2        = ?"
115 INPUT "            F2 = ";F2: PRINT AT 17,28;F2
117 PRINT AT 19,1;"-Angulo de aplicacion F2 = ?
(0-360)"
120 INPUT "            W2 = ";W2: PRINT AT 19,28;W2
130 CLS
140 REM
150 REM  TRAZADO DE LOS EJES
160 REM
165 CLS
170 FOR X=0 TO 255 STEP 2
180 PLOT  BRIGHT 1;X,88
190 NEXT X
200 FOR Y=0 TO 175 STEP 2
210 PLOT  BRIGHT 1;128,Y
220 NEXT Y
230 REM
240 REM OPTIMIZACION DE LA POSICION
250 REM
260 LET A=(F1/(F1+F2))*88
270 LET B=(F2/(F1+F2))*88
280 REM
290 REM CALCULO DE LOS EXTREMOS
300 REM
310 LET AX=INT (A*COS ((W1*PI)/180))
320 LET AY=INT (A*SIN ((W1*PI)/180))
330 LET BX=INT (B*COS ((W2*PI)/180))
```

```
340 LET BY=INT (B*SIN ((W2*PI)/180))
350 REM
360 REM REPRESENTACION FUERZAS
370 REM
380 PLOT 128,88: DRAW  BRIGHT 1;AX,AY
390 PLOT 128,88: DRAW  BRIGHT 1;BX,BY
400 LET RX=AX+BX
410 LET RY=AY+BY
420 PLOT 128,88: DRAW  BRIGHT 1;RX,RY
430 PLOT 128+AX,88+AY: DRAW RX-AX,RY-AY
440 PLOT 128+BX,88+BY: DRAW RX-BX,RY-BY
450 PRINT #1; PAPER 4; INK 2; BRIGHT 1; FLASH 1;" PULSE UNA
TECLA PARA CONTINUAR "
460 LET A$=INKEY$
470 IF A$="" THEN  GO TO 460
480 CLS
490 REM
500 REM CALCULO DE LA RESULTANTE
510 REM
520 LET W=W1-W2
530 LET W=180-W: LET W=(W/180)*PI
540 LET R=SQR (F1*F1+F2*F2-2*F1*F2*COS (W))
550 PRINT AT 1,11; PAPER 3; INK 6; BRIGHT 1;"RESULTADOS"
560 PRINT AT 6,1;"Fuerza 1 = ";F1;AT 6,20;"Angulo = ";W1
570 PRINT AT 10,1;"Fuerza 2 = ";F2;AT 10,20;"Angulo = ";W2
580 PRINT AT 16,5; INK 5; BRIGHT 1;"RESULTANTE = ";R
590 PRINT #0;"          OTRA VEZ (S/N) "
600 LET A$=INKEY$
610 IF A$="S" THEN  CLS : GO TO 10
620 IF A$="N" THEN  INPUT 0: GO TO 640
630 GO TO 600
640 PRINT #0; FLASH 1; PAPER 6; INK 4;"        FIN DEL PROG
RAMA        ": PAUSE 0
```

5. PASO : Lista de las principales variables

F1 = Primera fuerza
F2 = Segunda fuerza
R = Resultante
X = Se usa en el cálculo de ésta coordenada
Y = Se usa en el cálculo de ésta coordenada
A = Magnitud auxiliar de optimización
B = Magnitud auxiliar de optimización
W = W1-W2
W1 = Primer ángulo de aplicación
W2 = Segundo ángulo de aplicación

6. PASO : Descripción del programa

Línea 10-60 : Título

Línea 80-120 : Introducción de los datos de partida

Línea 130 : Borrado de la pantalla

Línea 140-220 : Trazado de los ejes coordenados

Línea 230-270 : Optimización de las fuerzas para que no excedan los límites de la pantalla.

Línea 280-340 : Cálculo de las coordenadas de los extremos de las fuerzas.

Línea 350-440 : Trazado en pantalla de las dos fuerzas. " de su resultante

Línea 450-470 : Bucle de espera

Línea 480 : Borrado de la pantalla

Línea 490-540 : Cálculo de las componentes de la resulte.

Línea 550-580 : Impresión de los resultados del problema

Línea 590 : Pregunta si se desea repetir el programa

Línea 600-630 : Se espera e identifica que caracter se ha tecleado

Línea 640 : Fin del programa

7. PASO : Resultados

En vista del esquema del 1. Paso, prescindiremos aquí de ofrecer resultados.

4.9 La Ley de Ohm

1. PASO : Presentación del problema

La Ley de Ohm se cumple en aquellos conductores eléctricos en los que la corriente crece proporcionalmente a la diferencia de potencial.

Esta Ley dice lo siguiente:

$$\frac{\text{dif. de pot.}}{\text{intensidad}} = \text{constante} = R$$

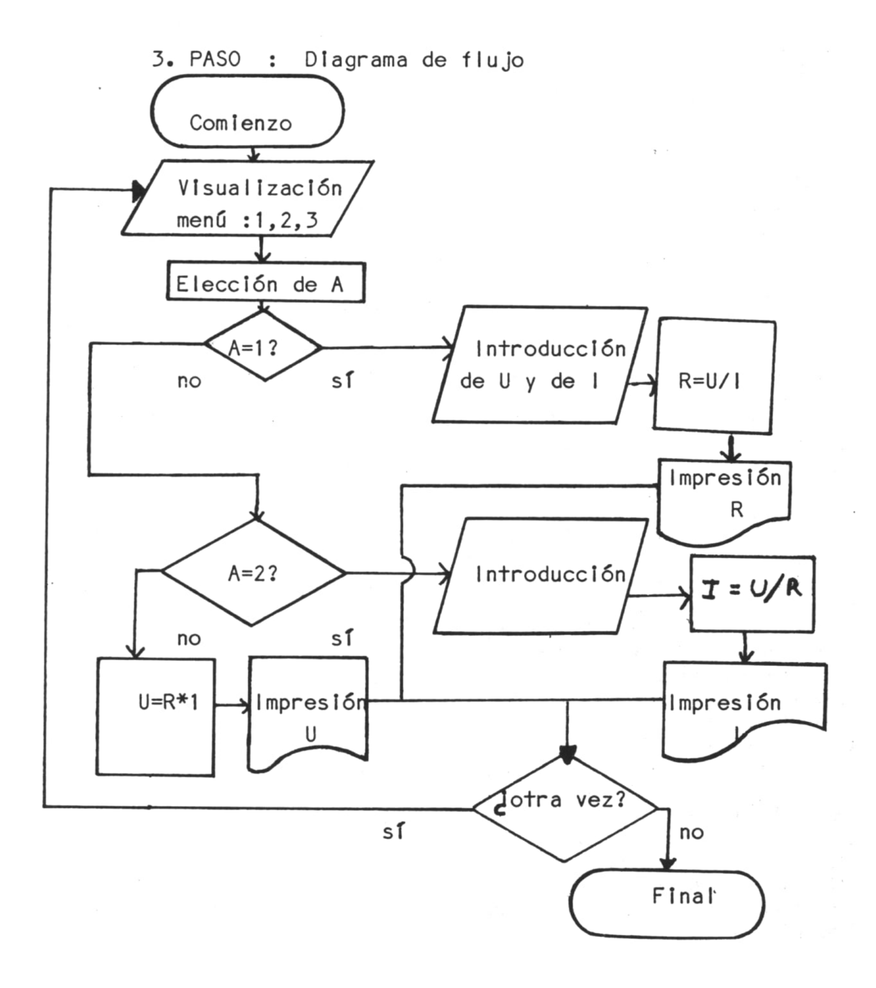
La magnitud R se denomina resistencia eléctrica del conductor.

En el programa siguiente se calcula para cada par de valores (dif. de pot., intensidad, resistencia), la tercera magnitud.

2. PASO : Análisis del problema

El programa que nos ocupa no ofrece ninguna dificultad desde el punto de vista matemático.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
  10 REM    F-6 LEY DE OHM
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA  MOSTRAR
LAS RELA-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"CIONES ENTRE LA RESISTE
NCIA,  LA"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"DIFERENCIA DE POTENCIAL
 Y LA IN-TENSIDAD.                          "
  50 PRINT : PRINT "    -----------------------  "
  55 PRINT AT 6,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    -----------------------  "
  65 PRINT AT 10,0; PAPER 1; BRIGHT 1;"*SELECCIONE LOS DATOS
 DE TRABAJO"
  66 PRINT "________________________________"
  70 PRINT AT 14,0;"DIF. DE POT. E INTENSIDAD....": PAPER 1;
 BRIGHT 1;"(1)"
  80 PRINT AT 16,0;"DIF. DE POT. Y RESISTENCIA...": PAPER 2;
 BRIGHT 1;"(2)"
  90 PRINT AT 18,0;"INTENSIDAD Y RESISTENCIA.....": PAPER 3;
 BRIGHT 1;"(3)"
 100 PRINT AT 21,6; FLASH 1;"INTRODUZCA UN NUMERO"
 105 FOR X=0 TO 50: BEEP .02,X: NEXT X
 110 LET A$=INKEY$
 120 IF A$="1" OR A$="2" OR A$="3" THEN  GO TO 130
 125 GO TO 110
 130 IF A$>"1" THEN  GO TO 190
 140 CLS : PRINT AT 4,4;"DIF. DE POTENCIAL = ?": INPUT "
      d.d.p. = ";U: PRINT AT 4,24;U;: PRINT " v."
 150 PRINT AT 8,4;"INTENSIDAD          = ?": INPUT "
 I. = ";I: PRINT AT 8,24;I;: PRINT " A."
 160 LET R=U/I: LET R=INT (R*100+.5)/100
 165 PRINT AT 12,0;"                              "
 170 PRINT AT 16,2;"RESISTENCIA        = ";R;: PRINT " ohms"
 180 GO TO 300
 190 IF A$="3" THEN  GO TO 250
 200 CLS : PRINT AT 4,3;"DIF. DE POTENCIAL = ?": INPUT "
    d.d.p. = ";U: PRINT AT 4,23;U;: PRINT " v."
 210 PRINT AT 8,3;"RESISTENCIA        = ?": INPUT "
  R. = ";R: PRINT AT 8,23;R;: PRINT " ohms"
 220 LET I=U/R: LET I=INT (I*100+.5)/100
 225 PRINT AT 12,0;"                              "
 230 PRINT AT 16,3;"INTENSIDAD         = ";I;: PRINT " A."
 240 GO TO 300
 250 CLS : PRINT AT 4,4;"RESISTENCIA       = ?";: INPUT "
        R. = ";R: PRINT AT 4,23;R;: PRINT " ohms"
```

```
260 PRINT AT 8,4;"INTENSIDAD        = ?";: INPUT "
I. = ";I: PRINT AT 8,23;I;: PRINT " A."
270 PRINT AT 12,0;"                              "
280 LET U=R*I: LET U=INT (U*100+.5)/100
290 PRINT AT 16,3;"DIF. DE POTENCIAL = ";U;: PRINT " v."
300 PRINT #0;"       MAS CALCULOS (S/N) "
310 LET A$=INKEY$
320 IF A$="S" THEN  CLS : GO TO 10
330 IF A$="N" THEN  INPUT O: GO TO 350
340 GO TO 310
350 PRINT #0; FLASH 1; PAPER 6; INK 4;"        FIN DEL PROG
RAMA         ": PAUSE 0
```

5. PASO : Lista de variables

A$ = Variable de selección (menú)
I = Intensidad
R = Resistencia
U = Diferencia de potencial

6. PASO : Descripción del programa

Línea 10-100 : Título y menú

Línea 105 : Bucle generador de sonido

Línea 110 : Espera la entrada de una de las opciones

Línea 120-125 : Solo acepta las opciones 1, 2 y 3

Línea 130 : Si A$ es mayor que 1, proseguir en la línea 190.

Línea 140-150 : Introducciones de acuerdo con A$=1

Línea 160-170 : Cálculos e impresión del resultado

Línea 180 : Salto a la línea 300

Línea 190 : Si se ha escogido la tercera opción se salta a la línea 250.

Línea 200-230 : Cálculo del valor de la intensidad

Línea 240 : Salto a la línea 300

Línea 250-290 : Cálculo análogo, pero para A$ = 3

Línea 300-340 : Consulta si se desean más cálculos
En caso afirmativo borrar la pantalla
y volver a la línea 10

Línea 350 : Final del programa

7. PASO : Resultados

Si seleccionamos el punto 1 del menú, el programa nos pedirá la diferencia de potencial y la intensidad. Si en respuesta introducimos, por ejemplo, el valor 220 para la diferencia de potencial y el valor 11 para la intensidad, obtendremos el siguiente resultado:

```
RESISTENCIA     :  20

MAS CALCULOS (S/N)
```

5

Capítulo 5 : Lenguas

5.1 Consideración previa y más BASIC

Los ordenadores pueden cumplir también útiles servicios para el aprendizaje y el repaso de idiomas extranjeros. No se trata aquí de presentar cursos completos de idiomas (por ejemplo, para aprender inglés), porque esto superaría ampliamente el marco y el nivel de este libro - además, estos "programas completos" pueden adquirirse ya en los comercios a precios relativamente asequibles. Nos interesa mucho más el confeccionar programas sencillos, que puedan servirle de ayuda al escolar a la hora de aprender idiomas.

Se consideran aquí en primera línea aquellos programas que facilitan el aprendizaje de vocablos o que proporcionan traducciones, ahorrándonos así la consulta de diccionarios.

No obstante, para este tipo de programas resultan necesarias algunas instrucciones BASIC nuevas.

A menudo no será útil poder separar determinados segmentos del programa principal para poder saltar a los mismos más de una vez.

Estos programas parciales se denominan subrutinas. El salto del programa principal a la subrutina requiere la siguiente instrucción:

Instrucción 22:

```
nn  GOSUB  mm
```

Esta instrucción provoca un salto a la subrutina que comienza en la línea mm.

Una vez recorrida la subrutina debe retornarse al programa principal y cerrar la subrutina mediante la instrucción siguiente:

Instrucción 23:

```
nn RETURN
```

Esta instrucción provoca un salto de vuelta a aquel punto del programa principal situado justo después de la correspondiente instrucción GOSUB.

El funcionamiento de ambas instrucciones queda patente en los siguientes programas, de modo que prescindiremos de cualquier ejemplo.

5.2 Verbos irregulares ingleses

1. PASO : Presentación del problema

En este primer ejemplo presentamos un programa que, tras introducir un verbo en castellano, ofrece su traducción al inglés y las correspondientes formas irregulares.

Así, por ejemplo tras introducir en el ordenador:

ir,

éste nos contesta

go went gone (ir).

Para que el programa no resulte demasiado largo, nos limitaremos a 20 verbos irregulares. El lector descubrirá de inmediato cómo debe completar el programa para obtener una panorámica sobre todos los verbos irregulares ingleses.

2. PASO : Análisis del problema

En este programa deben prepararse verbos castellanos, sus traducciones al inglés y las formas irregulares de estos verbos.

Cuando la cantidad de información es muy amplia, resulta aconsejable no leer las informaciones de partida tras cada comienzo del programa, sino efectuar la lectura interactiva de la información requerida a partir de un fichero grabado en un diskette convenientemente preparado.

Esta posibilidad, que seguramente utilizarán los programadores más adelantados - por lo menos si disponen de una unidad de diskettes - no vamos a tratarla aquí.

Así, tras cada inicio del programa haremos que el usuario indique el verbo acerca del cual quiere información. A renglón seguido, el programa buscará en el fichero de datos. Una vez hallado el verbo, podrá visualizarse el correspondiente registro.

Si el programa no hallase en su fichero el verbo introducido, sería debido a :

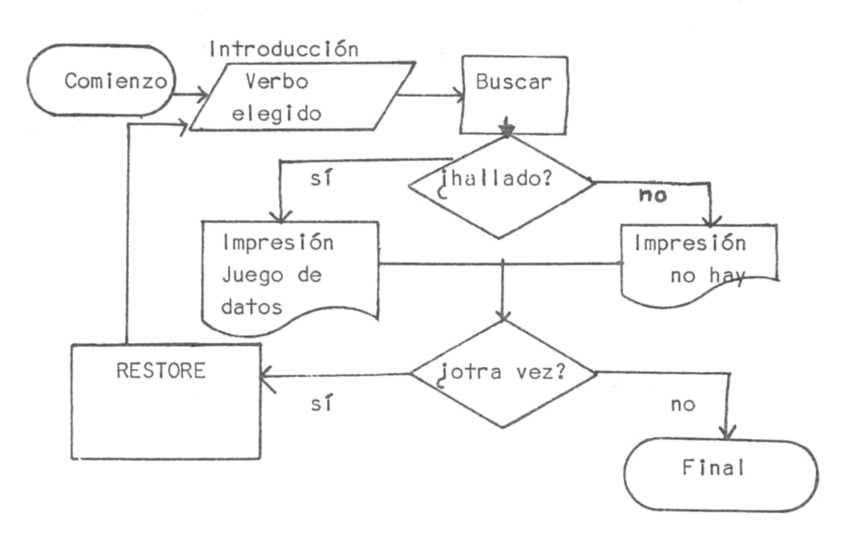
- o bien el verbo no existe aún en el archivo de datos (en principio solo hemos registrado 20 verbos)

- o bien el verbo introducido no es un verbo irregular.

En ambos casos, que debido a que la información de partida no es completa no pueden separarse el uno del otro, deberá visualizarse un mensaje.

En otro estado de cosas, el programa debe ofrecerle al usuario la posibilidad de consultar otro verbo.

3. PASO : Diagrama de flujo



```
  10 REM L1=VERBOS IRREG. ING.
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA TRADUCIR
AL INGLES"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"VERBOS ESPA/OLES.
      "
  50 PRINT : PRINT "    ------------------------   "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------   "
  70 LET A$="El numero de verbos existente eneste programa e
sta limitado. Pueden efectuarse ampliaciones, si se alarga l
a lista de DATAS (Li-nea 500 y sig.) y modificando enla line
a 160 el valor de N."
  80 PRINT : PRINT : PRINT
  90 FOR X=1 TO LEN A$
 100 BEEP .001,65: BEEP .09,30
 110 IF PEEK 23688=1 THEN  PRINT " "
 120 PRINT A$(X TO X);
 130 NEXT X
 140 PRINT #0; INK 2; BRIGHT 1; FLASH 1;" PARA PROSEGUIR PUL
SE UNA TECLA "
 145 LET a$=INKEY$
 146 IF a$="" THEN  GO TO 145
 150 CLS : RESTORE
 160 LET N=20
 170 PRINT AT 2,1;"INTRODUZCA EL VERBO ESPA/OL DEL"
 180 PRINT AT 4,1;"CUAL SE DESEAN SABER LAS FORMAS"
 190 PRINT AT 6,1;"INGLESAS."
 200 PRINT AT 9,10;"VERBO ? ": INPUT "              ";v$: PRI
NT AT 9,16;": ";v$
 210 FOR I=1 TO N
 220 READ j$: READ k$: READ l$: READ m$
 230 IF j$=v$ THEN  GO TO 260
 235 NEXT I
 240 PRINT AT 15,0; FLASH 1; INK 2; BRIGHT 1;"VERBO NO EXIST
ENTE EN EL FICHERO        O NO IRREGULAR.          "
 250 GO TO 310
 260 PRINT AT 11,9;"FORMAS INGLESAS"
 270 PRINT AT 11,9;"---------------"
 280 PRINT AT 15,14;k$
 290 PRINT AT 17,14;l$
 300 PRINT AT 19,14;m$
 310 PRINT #0;"          OTRA VEZ (S/N)"
 320 LET a$=INKEY$
 330 IF a$="S" THEN  CLS : GO TO 150
```

```
340 IF a$="N" THEN  INPUT O: GO TO 360
 350 GO TO 320
 360 PRINT #0; FLASH 1; PAPER 6; INK 4;"        FIN DEL PROG
RAMA         ": PAUSE 0
 500 DATA "SER","BE","WAS","BEEN"
 510 DATA "HABER","HAVE","HAD","HAD"
 520 DATA "COMENZAR","BEGIN","BEGAN","BEGUN"
 530 DATA "TRAER","BRING","BROUGHT","BROUGHT"
 540 DATA "VENIR","COME","CAME","COME"
 550 DATA "HACER","DO","DID","DONE"
 560 DATA "COMER","EAT","ATE","EATEN"
 570 DATA "COMPRAR","BUY","BOUGHT","BOUGHT"
 580 DATA "SENTIR","FEEL","FELT","FELT"
 590 DATA "ENCONTRAR","FIND","FOUND","FOUND"
 600 DATA "IR","GO","WENT","GONE"
 610 DATA "PERDER","LOOSE","LOST","LOST"
 620 DATA "HACER","MEKE","MADE","MADE"
 630 DATA "LEER","READ","READ","READ"
 640 DATA "CORRER","RUN","RAN","RUN"
 650 DATA "DECIR","SAY","SAID","SAID"
 660 DATA "VER","SEE","SAW","SEEN"
 670 DATA "SENTARSE","SIT","SAT","SAT"
 680 DATA "HABLAR","SPEAK","SPOKE","SPOKEN"
 690 DATA "COGER","TAKE","TOOK","TAKEN"
```

5. PASO : Lista de variables

A$: Variable de cadena para introducciones
D$: Verbo castellano
E1$: Verbo inglés, forma 1
E2$: Verbo inglés, forma 2
E3$: Verbo inglés, forma 3
I : Variable índice
N : Número de registros
V$: Verbo elegido

6. PASO : Descripción del programa

Línea 10-130 : Título, aclaraciones, indicaciones acerca de cómo debe actualizarse el archivo de datos

Línea 140-146 : Programa en ''posición de espera''. La realización del programa prosigue únicamente tras pulsar una tecla

Línea 150 : Borrado de la pantalla y restauración del puntero de DATAS a 0.

Línea 160 : Indicación del número de verbos

Línea 170-200 : Demanda del verbo elegido

Línea 210-235 : Busca el fichero de datos correcto, en caso de hallarlo sigue en 260; en caso de no hallarlo sigue en 240

Línea 240-250 : Mensaje para indicar que el verbo no existe en el fichero de datos; seguir en la línea 310.

Línea 260-300 : Impresión de las informaciones buscadas

Línea 310-350 : Consulta si se desea otra impresión; en caso afirmativo, borrado de la pantalla y vuelta a la línea 150

Línea 340 : En caso negativo final del programa

Línea 500-690 : DATAS

7. PASO : Resultados

Si introducimos por ejemplo en respuesta a la demanda del ordenador en la línea 200:

```
IR,
```

El ordenador contesta:

```
VERBO : IR

FORMAS INGLESAS :

     GO
     WENT
     GONE

OTRA VEZ (S/N) ?
```

Puede verse claramente cómo hay que modificar el programa para completar la lista de verbos irregulares : En la línea 500 y siguientes habrá que añadir más instrucciones DATA y en la línea 160 habrá que apuntar el nuevo número de verbos.

5.3 Vocablos franceses

1. PASO : Presentación del problema

En este ejemplo vamos a mostrar como puede encontrar aplicación el ordenador para sustituir la libreta de vocablos. Este contesta a la palabra francesa que le introducimos con el equivalente castellano o viceversa (rige aquí la misma observación del punto 5.2 acerca de la utilización interactiva de una memoria de diskettes).

Para que el programa no crezca demasiado, lo limitaremos a diez vocablos. Es decir, que sirve sólo a modo de ejemplo. Para aprovecharlo realmente, el lector deberá tener la paciencia de introducir previamente en el ordenador todo su archivo castellano-francés.

En los principios de funcionamiento del programa no hay ningún cambio.

2. PASO : Análisis del problema

Con este problema podemos ser escuetos:

Cuando el usuario introduce una palabra castellana, el ordenador debe buscar entre su archivo de datos el correspondiente par de palabras castellano-francesas mediante la comparación çon todos los vocablos castellanos (y sus correspondientes parejas en francés) de que dispone para, después, visualizarlas.

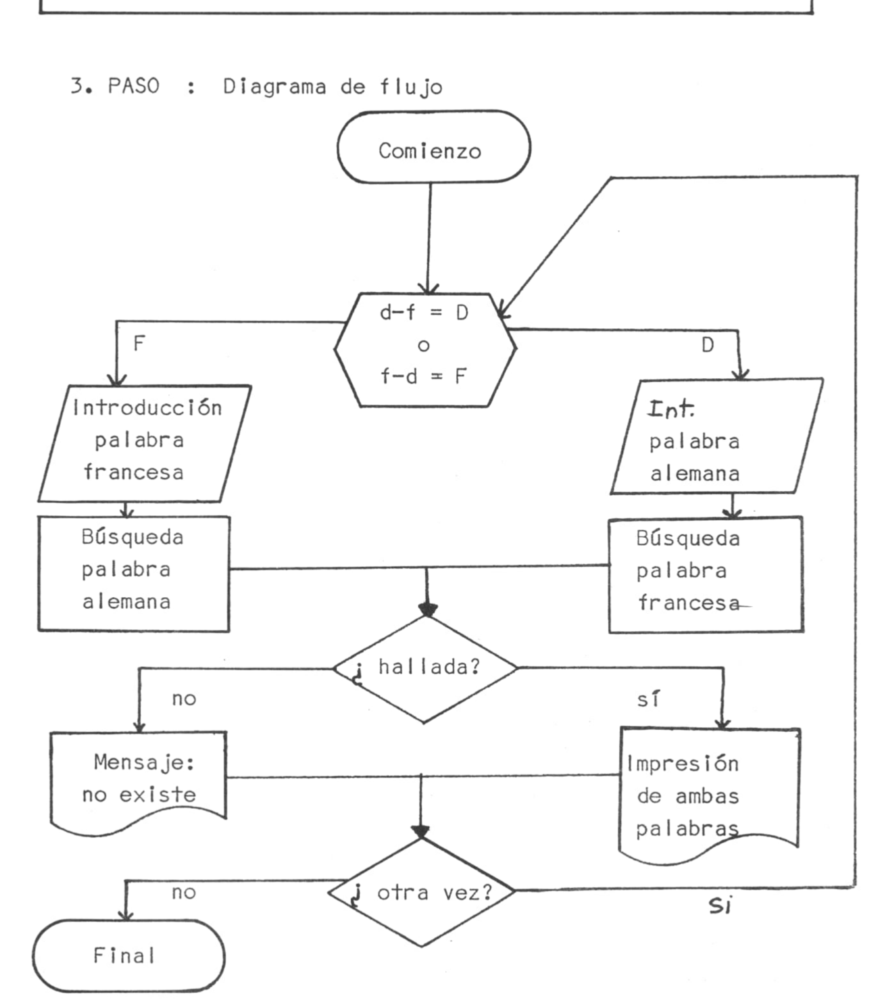
Para que ambos "sentidos" de la traducción puedan ser cubiertos por un solo programa, primero hay que decirle al ordenador si se desea la traducción del castellano al francés o del francés al castellano.

En caso de no poder traducir la palabra introducida por no estar incluida (aún) en la (muy) corta lista de vocablos, deberá visualizarse el correspondiente mensaje.

Además, debe ofrecérsele al usuario la posibilidad de obtener otra traducción en la ejecución del mismo programa, si así lo desea.

Voss	Çapítulo	5	:	Lenguas	Página
	Punto	3	:	Vocablos	191
Colegio					

3. PASO : Diagrama de flujo



```
  10 REM L2-VOCABLOS CASTELLANO-FRANCES
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA CONSULTAR
 VOCABLOS"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"CASTELLANO-FRANCES O VI
CEVERSA. "
  50 PRINT : PRINT "    ------------------------   "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------   "
  70 LET A$="El numero de vocablos existentesen este program
a esta limitado. Pueden efectuarse ampliaciones, si se alarg
a la lista de DATAS  (Linea 500 y sig.), y aumentandoen la l
inea 160 el valor de N."
  80 PRINT : PRINT : PRINT
  90 FOR X=1 TO LEN A$
 100 BEEP .001,65: BEEP .04,30
 110 IF PEEK 23688=1 THEN  PRINT " "
 120 PRINT A$(X TO X);
 130 NEXT X
 140 PRINT #0; INK 2; BRIGHT 1; FLASH 1;" PARA PROSEGUIR PUL
SE UNA TECLA "
 145 LET a$=INKEY$
 146 IF a$="" THEN  GO TO 145
 150 CLS : RESTORE
 160 LET N=10
 170 PRINT AT 2,6;"QUE TRADUCCION DESEA ?"
 180 PRINT AT 3,6; INK 3; BRIGHT 1;"______________________"
 190 PRINT AT 9,6;"CASTELLANO-FRANCES "; PAPER 1; BRIGHT 1;"
(1)"
 200 PRINT AT 12,16;"O"
 210 PRINT AT 15,6;"FRANCES-CASTELLANO "; PAPER 2; BRIGHT 1;
"(2)"
 220 PRINT #0; FLASH 1; BRIGHT 1; PAPER 2; INK 4;"      PULS
E 1 o 2, POR FAVOR    "
 230 LET A$=INKEY$
 240 IF A$="1" THEN  GO TO 270
 250 IF A$="2" THEN  GO TO 350
 260 GO TO 230
 270 CLS : PRINT AT 5,4;"PALABRA CASTELLANA - ?": INPUT TAB
11;"? ";P$: PRINT AT 5,25;P$
 280 FOR I=1 TO N
 290 READ C$: READ F$
 300 IF P$=C$ THEN  GO TO 330
 310 NEXT I
 320 GO TO 430
```

```
330 PRINT AT 12,4;"PALABRA FRANCESA   = "; PAPER 5;F$
340 GO TO 440
350 CLS : PRINT AT 5,4;"PALABRA FRANCESA   = ?": INPUT TAB
11;"? ";P$: PRINT AT 5,25;P$
360 FOR I=1 TO N
370 READ C$: READ F$
380 IF P$=F$ THEN  GO TO 410
390 NEXT I
400 GO TO 430
410 PRINT AT 12,4;"PALABRA CASTELLANA = "; PAPER 2; BRIGHT
1;C$
420 GO TO 440
430 PRINT AT 12,0; PAPER 2; INK 4; FLASH 1;" EL VOCABLO BUS
CADO NO SE HALLA           EN EL FICHERO           "
440 PRINT #0;"          OTRA VEZ (S/N)"
450 LET a$=INKEY$
460 IF a$="S" THEN  CLS : GO TO 150
470 IF a$="N" THEN  INPUT O: GO TO 490
480 GO TO 450
490 PRINT #0; FLASH 1; PAPER 6; INK 4;"        FIN DEL PROG
RAMA         ": PAUSE O
500 DATA "COMPRAR","ACHETER"
510 DATA "HABER","AVOIR"
520 DATA "IR","ALLER"
530 DATA "VER","VOIR"
540 DATA "BEBER","BOIRE"
550 DATA "HACER","FAIRE"
560 DATA "LEER","LIRE"
570 DATA "COMER","MANGER"
580 DATA "TOMAR","PRENDRE"
590 DATA "FUMAR","FUMER"
```

5. PASO : Lista de variables

A$ = Variable de cadena para introducciones

D$ = Vocablos castellanos

F$ = Vocablos franceses

I = Variable índice

N = Número de registros

W$ = Palabra elegida

6. PASO : Descripción del programa

Línea 10-130 : Título, aclaraciones etc.

Línea 140-150 : Espera, borrado de la pantalla e inicialización del puntero de DATAS.

Línea 160 : Indicación del número de líneas de datos

Línea 170-220 : Visualización del menú de opciones.

Línea 230-260 : Identificación de la opción elegida.

Línea 250 : Si se introduce 2 (francés-castellano), seguir en la línea 350

Línea 270 : Introducción de un vocablo castellano

Línea 280-310 : Busca el correspondiente juego de datos

Línea 320 : Si la palabra buscada no se halla en el archivo de datos y seguir en 430

Línea 330 : Impresión de los resultados

Línea 340 : Salto a la línea 440

Línea 350-430 : Procedimiento análogo para la introducción de la palabra francesa

Línea 440-490 : Finalización del programa con posibilidad de efectuar otra consulta

Línea 500-590 : DATAS

7. PASO : Resultados

Si pulsamos el número 1 en el menú, decidiéndonos por la traducción del castellano al francés, el ordenador nos pedirá una palabra castellana.

Si en respuesta introducimos la palabra, el ordenador imprimirá:

```
CASTELLANO   : TENER

FRANCES      : AVOIR

 OTRA VEZ  (S/N) ?
```

5.4 Test de vocablos ingleses

1. PASO : Presentación del problema

De forma similar al punto anterior, aquí deben ponerse a nuestra disposición traducciones a partir de un determinado fichero de vocablos. Pero, en contraposición al programa precedente, el usuario es requerido para contestar a una palabra castellana elegida al azar por el ordenador con la palabra inglesa correcta.

Es decir, que presentamos un programa que funciona del mismo modo que el aprendizaje de vocablos con la columna de palabras en el idioma extrajero tapada de antemano.

Puede observarse rápidamente que este programa es también aplicable a otros idiomas.

Con el fin de reducir el tiempo de programación nos limitaremos a ofrecer solo diez vocablos a modo de ejemplo. El ordenador selecciona en cada caso un subconjunto aleatorio del mismo, siendo informado el usuario acerca del número de aciertos por serie.

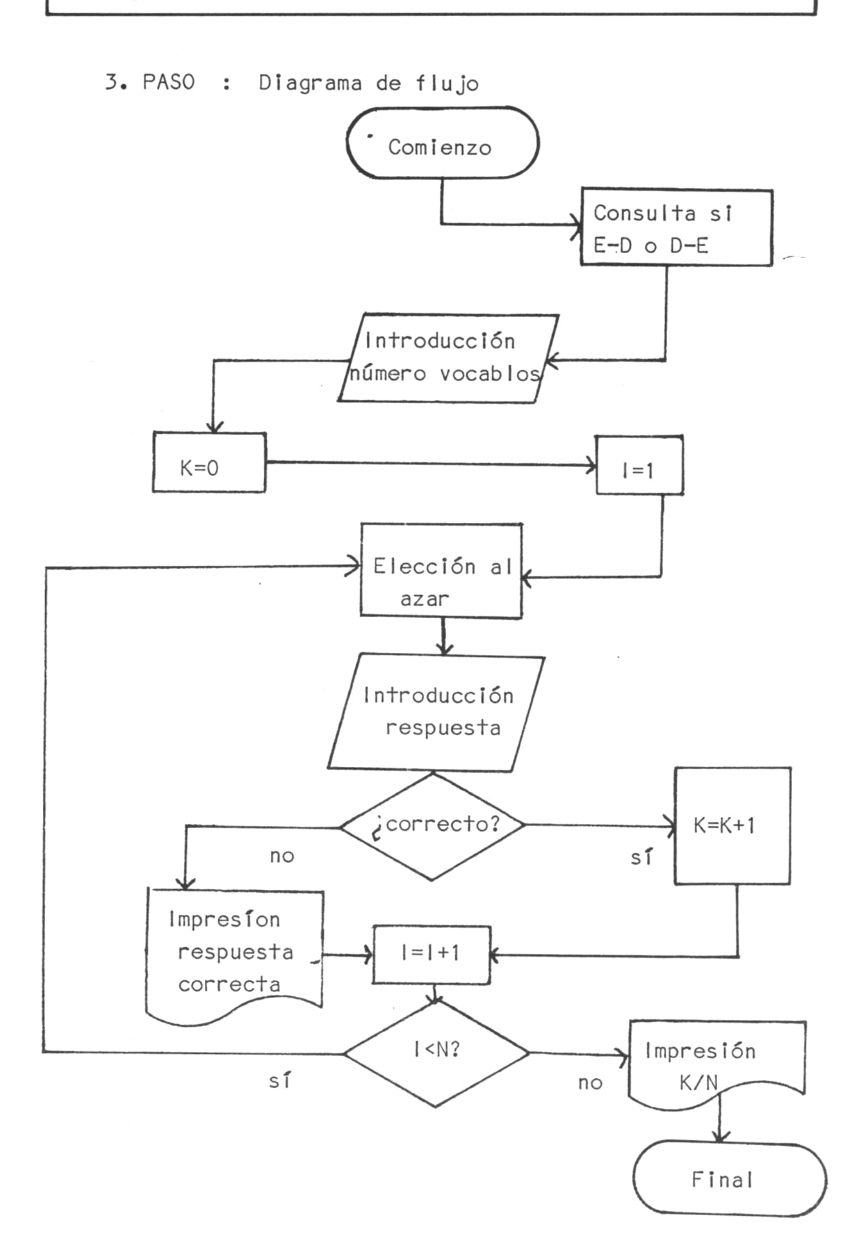
2. PASO : Análisis del problema.

Partiendo de un fichero de datos dado, el programa debe realizar una selección al azar.

El usuario responde a cada vocablo visualizado. El número de respuestas correctas es comparado con el número de preguntas.

El análisis del problema no ofrece ninguna otra dificultad.

3. PASO : Diagrama de flujo



```
  10 REM L3-TEST DE VOCABLOS
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"TEST DE VOCABLOS  CASTE
LLANO/IN-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"GLES O VICEVERSA.
       "
  50 PRINT : PRINT "    -----------------------  "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------  "
  70 LET A$="Este programa contiene solo -10-vocablos en su
fichero. Para am-pliar el programa a/ada mas sen-tencias  DA
TA a partir de la li-nea 500, y modifique en la linea160 el
valor de N."
  80 PRINT : PRINT : PRINT
  90 FOR X=1 TO LEN A$
 100 BEEP .001,65: BEEP .04,30
 110 IF PEEK 23688=1 THEN  PRINT " "
 120 PRINT A$(X TO X);
 130 NEXT X
 140 PRINT #0; INK 2; BRIGHT 1; FLASH 1;" PARA PROSEGUIR PUL
SE UNA TECLA "
 145 LET a$=INKEY$
 146 IF a$="" THEN  GO TO 145
 150 CLS : RESTORE
 160 LET N=10
 170 PRINT AT 2,8;"MENU DE OPCIONES"
 180 PRINT AT 3,8; INK 3; BRIGHT 1;"________________"
 190 PRINT AT 9,6;"CASTELLANO-INGLES "; PAPER 1; BRIGHT 1;"(
1)"
 200 PRINT AT 12,16;"O"
 210 PRINT AT 15,6;"INGLES-CASTELLANO "; PAPER 2; BRIGHT 1;"
(2)"
 220 PRINT #0; FLASH 1; BRIGHT 1; PAPER 2; INK 4;"      PULS
E 1 o 2, POR FAVOR    "
 230 LET A$=INKEY$
 240 IF A$="1" OR A$="2" THEN  GO TO 270
 260 GO TO 230
 270 CLS
 280 PRINT AT 10,0;"CUANTOS VOCABLOS QUIERE REPASAR?": INPUT
 TAB 4;"NUMERO DE VOCABLOS = ? ";A
 290 CLS
 300 LET K=0
 310 LET I=1
 320 LET R=INT (RND*N+1)
```

```
 330 FOR J=1 TO R
 340 READ D$: READ E$
 350 NEXT J
 360 IF A$="2" THEN  LET H$=D$: LET D$=E$: LET E$=H$
 370 PRINT AT 5,6;D$;"  EQUIVALE A:  ";
 380 FOR X=1 TO LEN (D$)
 390 PRINT AT 6,5+X;"-";
 400 NEXT X
 410 PRINT "-----------"
 420 PRINT AT 11,6;"RESPUESTA : ": INPUT TAB 10;W$: PRINT AT
 11,18;W$
 430 IF W$=E$ THEN  LET K=K+1: PRINT AT 18,11; FLASH 1; PAPE
R 2; INK 4; BRIGHT 1;;"MUY BIEN !": PAUSE 100: CLS : GO TO 4
60
 440 PRINT AT 15,5;"HA COMETIDO UN ERROR"
 450 PRINT AT 19,4;"LA RESPUESTA CORRECTA ES : ";AT 21,12; F
LASH 1;E$: PAUSE 400: CLS
 460 LET I=I+1
 470 IF I<=A THEN  RESTORE : GO TO 320
 480 LET KA=(K/A)*100: LET KA=INT (KA*100+.5)/100
 485 PRINT AT 10,1; PAPER 2; INK 4;"RESPUESTAS CORRECTAS : "
;KA;" %"
 490 PRINT #0; FLASH 1; PAPER 6; INK 4;"        FIN DEL PROG
RAMA        ": PAUSE 0
 500 DATA "CORRER","RUN","ESCRIBIR","WRITE"
 510 DATA "IMPRIMIR","PRINT","IR","GO"
 520 DATA "SI","IF","SUBRUTINA","SUBROUTINE"
 530 DATA "ORDENADOR","COMPUTER","CARACTER","CHARACTER"
 540 DATA "PANTALLA","SCREEN","TECLA","KEY"
```

5. PASO : Lista de variables

A = Número de vocablos a comprobar
A$ = Cadena de respuesta
D$ = Vocablos castellanos
E$ = Vocablos ingleses
I = Variable índice
J = Variable índice
K = Número de respuestas correctas
KA = Proporción de respuestas correctas
N = Número de vocablos en el fichero
R = Número aleatorio
W$ = Respuesta

6. PASO : Descripción del programa

Línea 10-130 : Título, aclaraciones y recordatorio de la eventual ampliación del programa

Línea 140-146 : Espera una entrada desde el teclado

Línea 150 : Borrado de la pantalla y puesta a 0 del puntero de DATAS.

Línea 160 : Indicación del número de vocablos incluidos en el fichero de datos.

Línea 170-270 : Impresión del menú e identificación de la opción elegida.

Línea 280-290 : Demanda del número de vocablos a repasar

Línea 300-310 : Contadores K e I a las posiciones iniciales

Línea 320 : Generación de un número aleatorio

Línea 330-350 : Lectura de datas

Línea 360 : Eventual cambio de idioma

Línea 370-410 : Presentación de la pregunta

Línea 420 : Petición de una respuesta

Línea 430-450 : Evaluación de la respuesta. Si es correcta se imprime mensaje de felicitación, si no la es, se sigue en la línea 440 y se imprime la respuesta correcta.

Línea 460-470 : Vocablo siguiente

Línea 480-485 : Cálculo e impresión de la cuota de aciertos

Línea 490 : Fin del programa

Línea 500-540 : DATAS

7. PASO : Resultados

No hace falta ofrecer aquí los resultados. Aquellos que son visualizados durante el transcurso del programa resultan directamente de la descripción del mismo

5.5 Escritura en clave

1. PASO : Presentación del problema

Después de los ejemplos anteriores, que eran típicamente''escolares'', para variar un poco presentamos un programa de carácter más bien lúdico - la creación de una escritura en clave.

Este programa convierte tantas frases como queramos a través de un código, en frases incomprensibles.

Se trata simplemente de convertir los caracteres utilizados normalmente para escribir, en otros caracteres siguiendo una regla que debe fijarse. El usuario podrá manipular la conversión de la forma que desee.

2. PASO : Análisis del problema

El análisis del problema no es excesivamente complicado:
Se trata únicamente de enfrentar el juego de caracteres usual (A,B,C...,inclusive los símbolos especiales que queramos utilizar) al "juego de caracteres secreto". Ambas cosas pueden hacerse con variables de cadena adecuadas.

Después de esto, la frase introducida debe ser codificadade carácter a carácter.
Para hacer factible esto deberemos utilizar dos funciones, hasta ahora no mencionadas: La primera función sirve para determinar la longitud de una cadena, es decir, el número de caracteres de que consta (incluídos los espacios en blanco).

La segunda función sirve para "arrancar" caracteres sueltos de una cadena dada.

La función citada en primer lugar requiere el comando LEN (LEN = length = longitud).
Por ejemplo, si el usuario introduce estando en la modalidad directa:

```
PRINT  LEN ("COMPUTER")
```

el ordenador contestará:

```
8
```

Esta función nos ha servido para determinar que la cadena "computer" consta de ocho caracteres.

Una subcadena es un grupo de caracteres consecutivos extraídos de una cadena. Por ejemplo: "comp" es una subcadena de "computer", sin embargo, "cmpu" no lo es, ya que sus caracteres no han sido extraídos de forma consecutiva.

Para extraer una subcadena se emplea la operación de fragmentación. La sintaxis general es:

Subcadena = Cadena de origen (a TO b)

En donde a es el número de orden del primer carácter a extraer, y b, el último, entre ambos números queda limitada la longitud de la subcadena. Existen tres tipos de fragmentación:

1.- Fragmentación de un solo carácter:

Carácter = Cadena de origen (n)

Donde n es el número de orden del carácter a extraer.

2.- Fragmentación izquierda:

Subcadena = Cadena de origen (TO n)

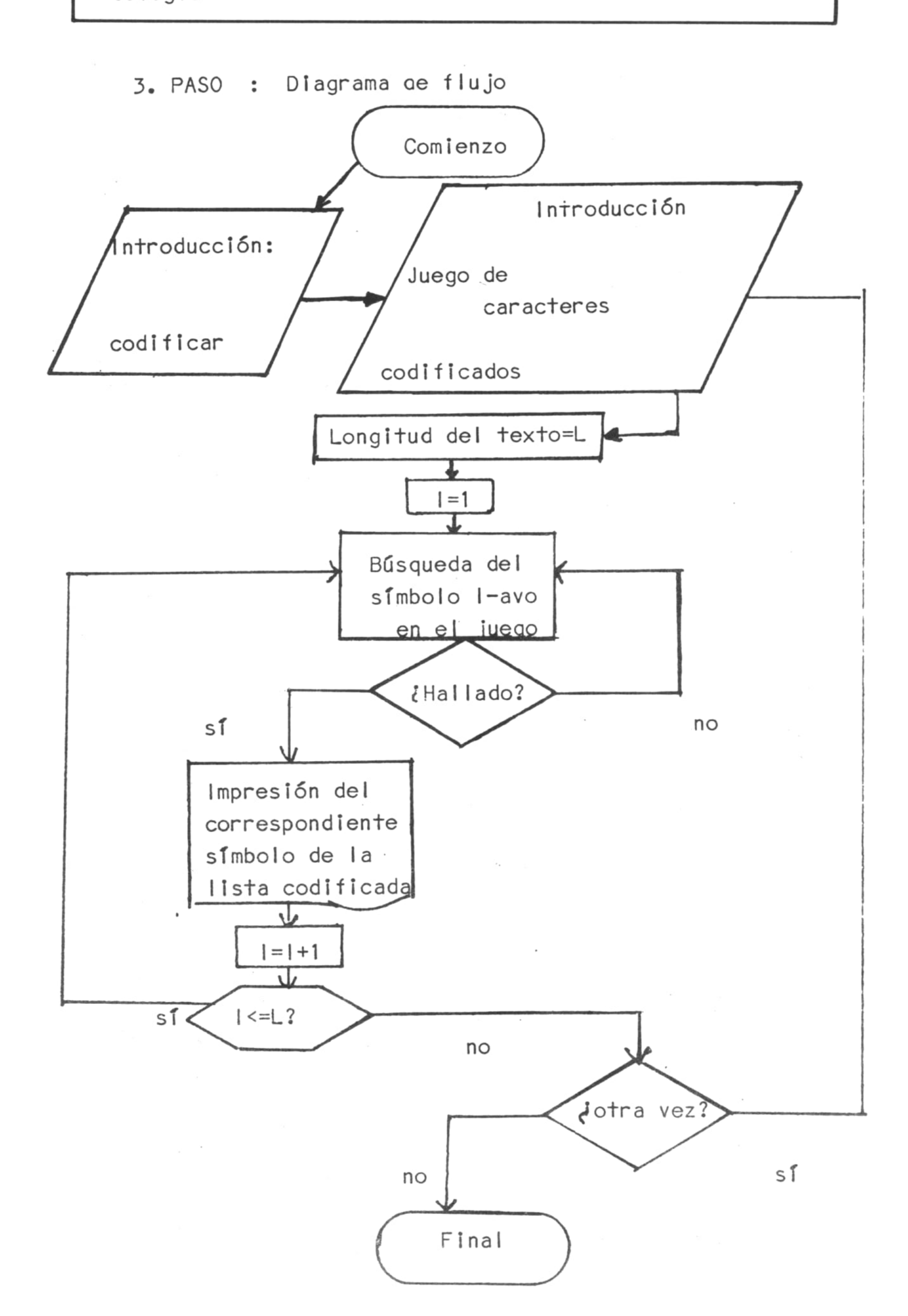
Consiste en la extracción de los n primeros caracteres de una cadena.

3.- Fragmentación derecha:

Subcadena = Cadena de origen (n TO)

Consiste en la extracción de los últimos caracteres de una cadena a partir de n.

3. PASO : Diagrama de flujo



```
  10 REM L4-ESCRITURA EN CLAVE
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;" PROGRAMA PARA CIFRAR C
UALQUIER "
  40 PRINT  PAPER 4; INK 0; FLASH 1;" TEXTO.
       "
  50 PRINT : PRINT "     ------------------------   "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------  "
  70 LET A$=" Para modificar la clave, cambiela linea 170.
                  Para mantener en secreto el co-digo defini
r la cadena C$ mediante una sentencia INPUT, cuyo contenido
 no debe ser dado a cono-cer."
  80 PRINT : PRINT : PRINT
  90 FOR X=1 TO LEN A$
 100 BEEP .001,65: BEEP .04,30
 110 IF PEEK 23688=1 THEN  PRINT " "
 120 PRINT A$(X TO X);
 130 NEXT X
 140 PRINT #0; INK 2; BRIGHT 1; FLASH 1;" PARA PROSEGUIR PUL
SE UNA TECLA "
 145 LET a$=INKEY$
 146 IF a$="" THEN  GO TO 145
 150 CLS
 160 LET B$="ABCDEFGHIJKLMNOPQRSTUVWXYZ ,."
 170 LET C$="QWERTYUIOPASDFGHJKL ZX,C.VBNM"
 180 PRINT AT 5,1;"INTRODUCE POR FAVOR EL TEXTO A"
 190 PRINT AT 7,10;"CODIFICAR :"
 200 INPUT "? ";A$
 210 LET L=LEN A$
 215 PRINT : PRINT : PRINT
 220 FOR I=1 TO L
 230 FOR J=1 TO LEN B$
 240 IF A$(I TO I)<>B$(J TO J) THEN  GO TO 270
 250 PRINT C$(J TO J);
 260 GO TO 280
 270 NEXT J
 280 NEXT I
 290 PRINT #0;TAB 8; FLASH 1; PAPER 6; INK 4;" OTRA VEZ (S/N
) "
 300 LET A$=INKEY$
 310 IF A$="S" THEN  GO TO 150
 320 IF A$="N" THEN  GO TO 340
 330 GO TO 300
 340 INPUT 0: PRINT #0; FLASH 1; PAPER 6; INK 1; BRIGHT 1;"
      FIN DEL PROGRAMA         ": PAUSE 0
```

5. PASO : Lista de variables

A$ = Texto a codificar
B$ = Juego de caracteres
C$ = Carácter cifrado
I = Indice variable
J = Indice variable
L = Longitud del texto a codificar
Z$ = Cadena de respuesta (S/N)

6. PASO : Descripción del programa

Línea 10-140 : Impresión del título y aclaraciones

Línea 145-150 : Interrupción del programa y borrado de la pantalla

Línea 160 : Definición del juego de caracteres

Línea 170 : Definición clave

Línea 180-200 : Demanda del texto a codificar

Línea 210 : Determinación de la longitud del texto

Línea 220-280 : Codificación

220 : Comienzo del bucle I (para todos los caracteres del texto)

230 : Comienzo del bucle J (para el juego de caracteres)

240 : Verificación de que cada carácter del texto se corresponde con un carácter del juego;
en caso negativo, seguir en 270
en caso afirmativo, impresión del correspondiente carácter cifrado

260 : Salto hacia la línea 280, es decir, proceso del siguiente carácter del texto
270 : Comprobación del siguiente carácter del juego de caracteres
280 : Siguiente carácter del texto

Línea 290-330 : Consulta si se desea una nueva codificación

Línea 340 : Final del programa

7. PASO : Resultados

Los resultados de este programa dependen de la directriz seguida para codificar (ver línea 170 : C$) y del texto a codificar (ver línea 200 : A$).

Si introducimos p.e. en el programa el texto WERNER VOSS obtendremos:

,TKFTKBXGLL

6

Capítulo 6 : Biología/Ecología

6.1 Aspectos previos

La aplicación del ordenador para problemas biológico-ecológicos se refiere especialmente a la simulación de procesos de desarrollo y de crecimiento (naturalmente pueden confeccionarse también programas de consulta siguiendo el patrón de los del capítulo anterior). Estos desarrollos pueden visualizarse en la pantalla gráficamente o en forma de tabla.

6.2 Crecimiento exponencial

1. PASO : Presentación del problema

Una población inicial de X0 individuos tiene una tasa de crecimiento anual del R por ciento (R debe ser positivo). Si este crecimiento no choca con limitaciones externas - cosa muy improbable a largo plazo - se produce un llamado crecimiento exponencial.

El desarrollo de un proceso de crecimiento tan sencillo como éste, se simulará con un programa BASIC.

2. PASO : Análisis del problema

Para este programa hay que introducir la población inicial y la tasa de crecimiento anual.

Después de un año se obtiene la población X1:

X1 = X0 + X0*R/100 = X0*(1 + R/100)

Después de dos años se obtiene una X2:

X2 = X1*(1 + R/100) = X0*(1 + R/100)*(1 + R/100)

De lo que resulta:

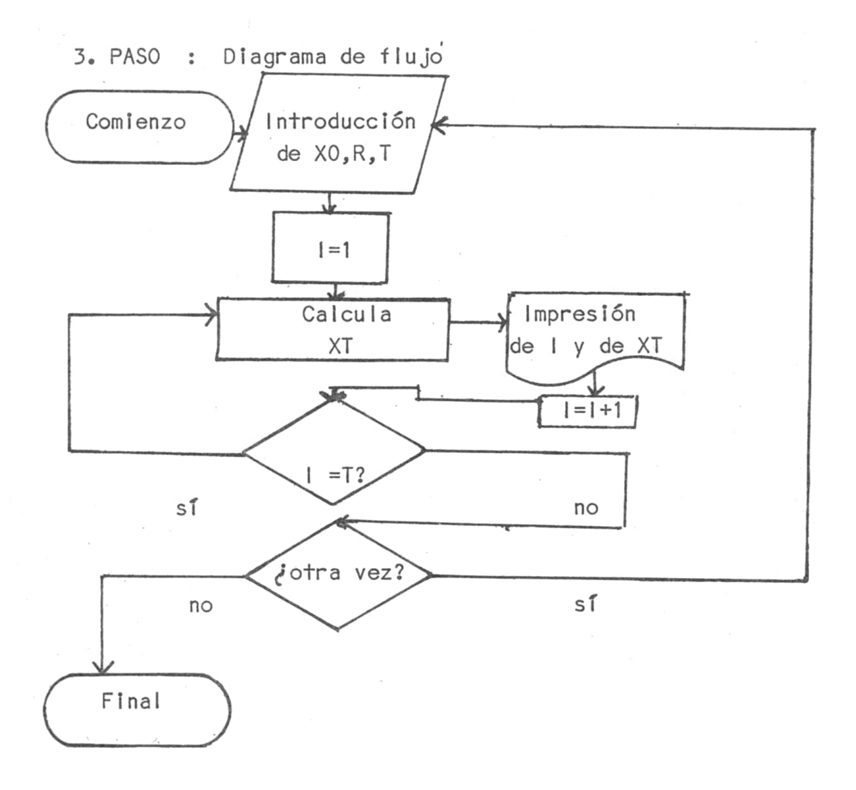
$$X2 = X0 * (1 + R/100)^2$$

Generalizando, después de T años se cumple que:

$$XT = X0 * (1 + R/100)^T$$

Esta fórmula indica el algoritmo de cálculo para el correspondiente programa BASIC.

3. PASO : Diagrama de flujo



```
  10 REM E1-CRECIMIENTO
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA REPRESENT
AR EN FOR"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"MA DE TABLA EL CRECIMIE
NTO EXPO-"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"NENCIAL.
        "
  50 PRINT : PRINT "    ------------------------  "
  55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------  "
  70 PRINT AT 11,0;"POBLACION INICIAL......: ";: INPUT TAB 1
0;"? ";X0: PRINT X0
  80 PRINT AT 15,0;"TASA DE CRECIMIENTO  % : ";: INPUT TAB 1
0;"? ";R: PRINT R
  90 PRINT AT 19,0;"NUMERO DE A/OS.........: ";: INPUT TAB 1
0;"? ";T: PRINT T
 100 CLS
 110 PRINT AT 0,6; FLASH 1; PAPER 1; INK 6;"A/OS";AT 0,17;"P
OBLACION"
 120 PRINT AT 1,6;"----";AT 1,17;"---------"
 130 LET I=1
 140 LET XT=X0*(1+R/100)^I
 150 LET XT=INT (XT*100+.5)/100
 160 PRINT
 170 PRINT "       ";I;"           ";XT
 180 LET I=I+1
 190 IF I/10=INT (I/10) THEN  PRINT #0; FLASH 1; PAPER 4; IN
K 2;" PULSE UNA TECLA PARA CONTINUAR ": PAUSE 0: CLS
 200 IF I<=T THEN  GO TO 140
 210 PRINT #0;TAB 8; FLASH 1; PAPER 6; INK 4;" OTRA VEZ (S/N
) "
 220 LET A$=INKEY$
 230 IF A$="S" THEN  GO TO 10
 240 IF A$="N" THEN  GO TO 260
 250 GO TO 220
 260 INPUT 0: PRINT #0; FLASH 1; PAPER 6; INK 1; BRIGHT 1;"
      FIN DEL PROGRAMA         ": PAUSE 0
```

5. PASO : Lista de variables

A$ = Cadena de respuestas (si/no)

I = Variable índice

R = Tasa de crecimiento en %

X0 = Población inicial

XT = Población en el momento T

6. PASO : Descripción del programa

Línea 10-60 : Título

Línea 70-90 : Visualización de las informaciones INPUT que necesita el programa

Línea 100-120 : Impresión del título de la tabla

Línea 130 : Preselección del primer período

Línea 140-150 : Cálculo de la población

Línea 160-170 : Impresión

Línea 180 : Paso al siguiente período

Línea 190 : Después de cada diez períodos se interrumpe el programa (es decir, cuando I es divisible entre 10); la continuación sólo es posible si se pulsa una tecla.

Línea 200 : Mientras I sea menor o igual a T, el programa debe continuar (línea 140)

Línea 210-260 : Final del programa, a menos que se pida un proceso más (entonces vuelta a la línea 20 después de borrar la pantalla)

7. PASO : Resultados

Si por ejemplo introducimos el valor 500 para la población inicial y una tasa de crecimiento del 4(%), además de un período de valor 8 (p.e. años), el programa genera la siguiente impresión en pantalla:

```
AÑOS     POBLACION
---      ---------

  1         520
  2         540.8
  3         562.43
  4         584.93
  5         608.33
  6         632.66
  7         657.97
  8         684.28

  OTRA VEZ   (S/N)
```

6.3 Crecimiento limitado

1. PASO : Presentación del problema

El crecimiento exponencial que ya presentamos en el punto anterior, no se da en la práctica, y si se da será por espacios de tiempo muy cortos. Deben considerarse ante todo efectos inhibidores, que serán más acentuados cuanto mayores sean los valores de las variables en juego.

Pensemos por ejemplo que una población con crecimiento exponencial choca muy pronto con limitaciones de nutrición, que dificultan más y más su posterior desarrollo.

Este tipo de evolución ''amortiguada'' va a ser simulada en el próximo programa BASIC.

2. PASO : Análisis del problema

El análisis del problema lleva consigo una forma de resolución similar a la del punto anterior:

También aquí se requieren valores de partida:

XO = Población de partida
RO = Tasa de crecimiento anual (en %) al comienzo del proceso de desarrollo

En contraposición al ejemplo precedente, no partimos de la base de que la tasa de crecimiento es constante, sino que provocamos su reducción a medida que aumenta la población.

Como la población aumenta con el paso del tiempo, mediante un esquema mental muy sencillo podemos deducir que la tasa de crecimiento se reduce progresivamente. Por ejemplo, podemos dividir entre dos la tasa cuando el tiempo se ha duplicado, dividirla entre cuatro cuando el tiempo se ha cuatriplicado, etc.

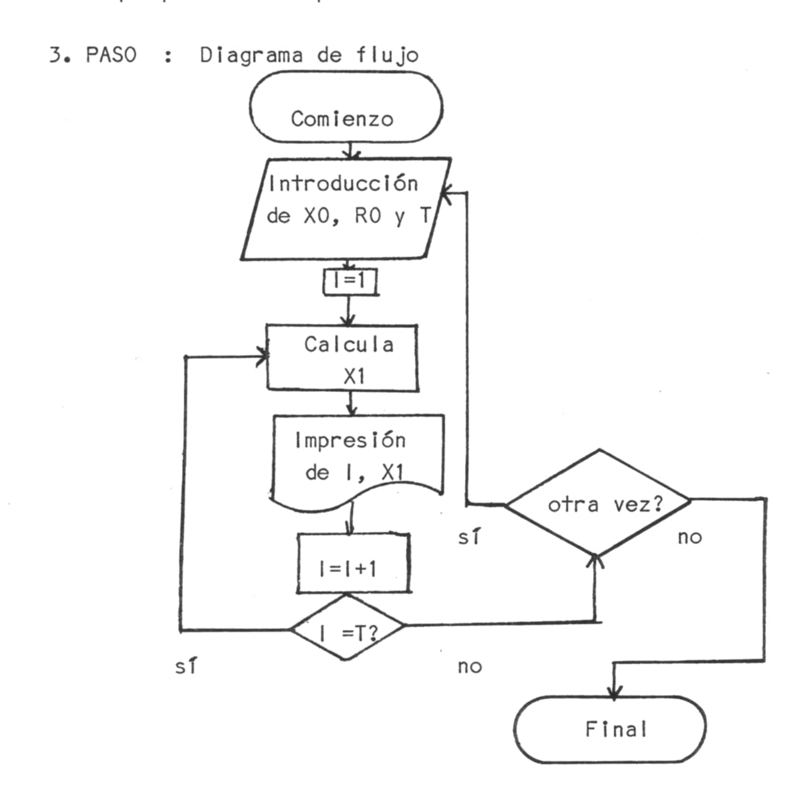
Claro que podemos establecer otros modelos matemáticos para simular este proceso de frenado, aunque esto no es de vital importancia dentro de este contexto.

El algoritmo que acabamos de describir puede expresarse matemáticamente de la forma siguiente:

X1 = X0 * (1 + R/I)

Es decir, en el período X1 la población se obtiene a partir de la del período precedente, como en el ejemplo anterior; en cambio, la tasa de crecimiento se divide entre el índice variable I, de manera que aquélla se reduce a medida que aumenta I, es decir, a medida que pasa el tiempo.

3. PASO : Diagrama de flujo



```
 310 PRINT AT 9,4;C$;" :"
 320 INPUT TAB 12;"? ";X$: PRINT AT 9,19;X$
 330 IF X$=S$ THEN  PRINT AT 17,9; FLASH 1; PAPER 5; INK 0;"
C O R R E C T O": PAUSE 250: GO TO 410
 340 PRINT AT 14,5; PAPER 2;"LA RESPUESTA ES ERRONEA": BEEP
3,-20
 350 PRINT #0;TAB 7;"OTRO INTENTO (S/N)"
 360 LET A$=INKEY$
 370 IF A$="S" THEN  CLS : GO TO 300
 380 IF A$="N" THEN  GO TO 400
 390 GO TO 360
 400 PRINT AT 18,5;"LA RESPUESTA CORRECTA ES : ";AT 20,11; P
APER 4; FLASH 1; INK 6;S$: PAUSE 250
 410 NEXT I
 420 INPUT 0: PRINT #0;TAB 8;"UN TEST MAS (S/N)"
 430 LET A$=INKEY$
 440 IF A$="S" THEN  GO TO 150
 450 IF A$="N" THEN  GO TO 470
 460 GO TO 430
 470 INPUT 0: PRINT #0; FLASH 1; PAPER 6; INK 4;"        FIN
 DEL PROGRAMA          ": PAUSE 0
 500 DATA "CATALU/A","BARCELONA","MADRID","MADRID"
 510 DATA "RIOJA","LOGRO/O","BALEARES","PALMA DE MALLORCA"
 520 DATA "PAIS VALENCIANO","VALENCIA","EUZKADI","GASTEIZ"
 530 DATA "CANTABRIA","SANTANDER","ANDALUCIA","SEVILLA"
 540 DATA "GALICIA","LA CORU/A","ASTURIAS","GIJON"
 550 DATA "EXTREMADURA","CACERES","CANARIAS","LAS PALMAS"
 560 DATA "MURCIA","MURCIA","CASTILLA LA MANCHA","CIUDAD REA
L"
 570 DATA "CASTILLA-LEON","VALLADOLID","ARAGON","ZARAGOZA"
 580 DATA "ASTURIAS","OVIEDO","NAVARRA","PAMPLONA"
```

5. PASO : Lista de variables

A$ = Cadena de respuesta (si/no)

I = Variable índice

R = Tasa de crecimiento al comienzo del proceso

T = Duración del proceso

X0 = Población inicial

X1 = Población después de un periodo

Z = Modificación de la población

6. PASO : Descripción del programa

Línea 10-60 : Título y aclaraciones

Línea 70-100 : Demanda de los datos de partida y borrado de la pantalla

Línea 110-120 : Impresión del título de la tabla

Línea 130 : Fijación del contador de períodos

Línea 140-170 : Cálculo de la población, del crecimiento e impresión de los valores tras redondear a dos posiciones decimales

Línea 180 : X1 toma el valor de X0 y se incrementa el contador.

Línea 190 : Se controla que el número de datos en pantalla no supere el número de 10

Línea 200 : Se evalua si deben continuarse los cálculos (mientras I siga siendo más pequeño que T después de sumarle 1)

Línea 210-260 : Consulta si todavía se desean más cá culos, en caso afirmativo vuelta a 10, y en caso negativo, fin del programa.

7. PASO : Resultados

Si introducimos como población inicial el valor 500, y una tasa de crecimiento en el primer año del 4%, además de indicar que el proceso debe durar 8 años, obtendremos:

AÑOS	POBLACION	CRECIMIENTO
1	520	4
2	530.4	2
3	537.47	1.33
4	542.84	1
5	547.18	.8
6	550.83	.67
7	553.98	.57
8	556.75	.5

OTRA VEZ (S/N)

6.4 Contaminación ambiental

1. PASO : Presentación del problema

Las poblaciones en crecimiento, por regla general frenan ellas mismas su desarrollo, debido a que deterioran sus condiciones de vida en grado creciente.

Un ejemplo típico de ello es la contaminación ambiental creciente a lo largo de las diversas fases de industrialización.

Es decir, aquí también se produce un "frenado del crecimiento" - similar al del punto anterior - que modifica el crecimiento exponencial que conocemos del punto 6.2.

En contraposición al punto anterior, este "frenado" no opera de acuerdo con el tamaño de la población propiamente dicho, sino de forma indirecta en dependencia de una variable que, a su vez, es proporcional a la población.

2. PASO : Análisis del problema

En el primer período, la población resultante del período 0 ya no se calcula

X1 = X0 + X0 * R/100

sino:

X1 = X0 + X0 * (R/100 - P*B0)

B0 es el factor contaminación ambiental en el momento 0, que frena el crecimiento de la población por medio de un parámetro P adecuado (p.e. dentro del ámbito de P = 0.01).

Lógicamente, necesitamos también una relación funcional que pueda expresar B1 (contaminación ambiental en el período 1) en dependencia de B0 y de X0.

Un posible ejemplo de relación sería:

B1 = B0 + A * X0

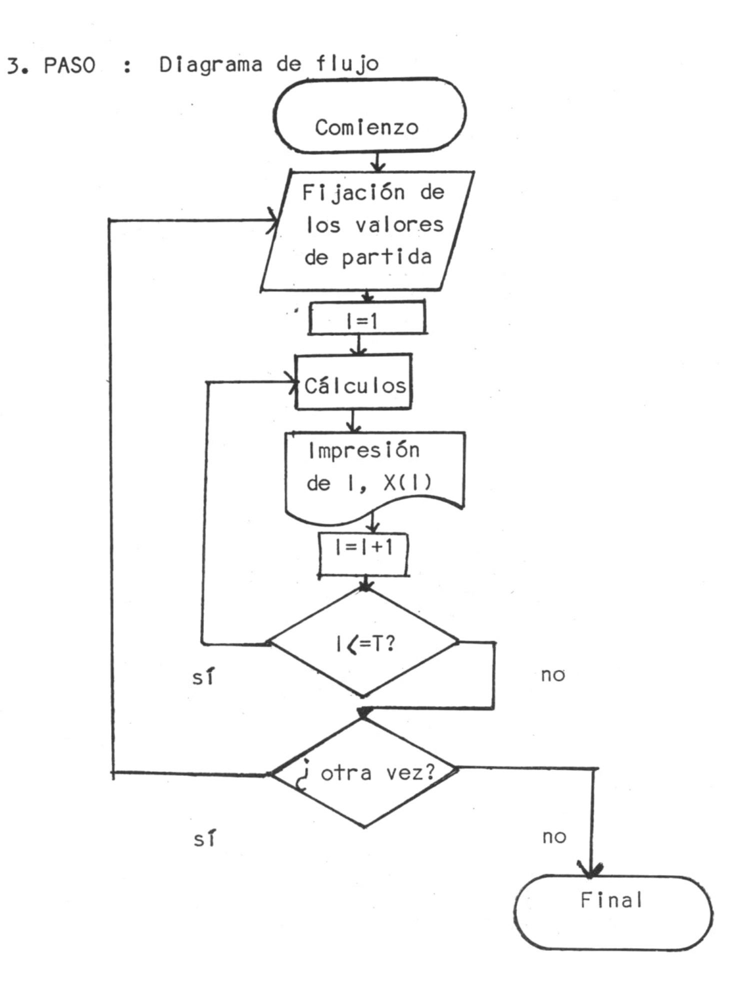
Partiendo de un factor B0 = 0, la contaminación aumenta a medida que crece la población, de acuerdo con un parámetro A, que debe dimensionarse adecuadamente (p.e. A = 0.015).

De este modo se cumple el siguiente algoritmo:

(1) B1 = B0 + A*X0

(2) X1 = X0 + X0 * (R/100 - P*B1)

3. PASO : Diagrama de flujo



```
 10 REM E3-CONT. AMBIENTAL
 20 BORDER 0: PAPER 0: INK 7: CLS
 25 POKE 23658,8: POKE 23609,35
 30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA SIMULAR E
N UNA TA-"
 40 PRINT  PAPER 4; INK 0; FLASH 1;"BLA, UN PROCESO DE  CRE
CIMIENTO "
 45 PRINT  PAPER 4; INK 0; FLASH 1;"FRENADO POR SU PROPIA D
INAMICA. "
 50 PRINT : PRINT "    -----------------------   "
 55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
 60 PRINT "    -----------------------   "
 70 PRINT AT 9,0;"POBLACION INICIAL......: ";: INPUT TAB 10
;"? ";X0: PRINT X0
 80 PRINT AT 13,0;"TASA DE CRECIMIENTO  % : ";: INPUT TAB 1
0;"? ";R: PRINT R
 90 PRINT AT 17,0;"NUMERO DE A/OS.........: ";: INPUT TAB 1
0;"? ";T: PRINT T
 95 PRINT AT 21,0;"FACTOR CONT. AMBIENTAL.: ";: INPUT TAB 1
0;"? ";P: PRINT P
100 CLS
110 PRINT AT 0,0; FLASH 1; PAPER 1; INK 6;"A/OS";AT 0,7;"PO
BLACION";AT 0,20;"^ %";AT 0,26;"CONT."
120 PRINT AT 1,0;"----";AT 1,7;"---------";AT 1,20;"---";AT
 1,26;"-----"
130 LET I=1: LET A=.015: LET B0=0
140 LET B1=B0+A*X0: LET X1=X0+X0*(R/100-P*B0)
150 LET Z=(X1-X0)/X0: LET Z=Z*100: LET X1=INT (X1*100+.5)/1
00: LET B1=INT (B1*100+.5)/100: LET Z=INT (Z*100+.5)/100
155 IF X1<=0 THEN  PRINT : PRINT  FLASH 1; PAPER 2; INK 4;"
     POBLACION ELIMINADA       ": GO TO 210
160 PRINT
170 PRINT TAB 1;I;TAB 9;X1;TAB 19;Z;TAB 27;B1
180 LET I=I+1
190 IF I/10=INT (I/10) THEN  PRINT #0; FLASH 1; PAPER 4; IN
K 2;" PULSE UNA TECLA PARA CONTINUAR ": PAUSE 0: CLS
200 IF I<=T THEN  LET B0=B1: LET X0=X1: GO TO 140
210 PRINT #0;TAB 8; FLASH 1; PAPER 6; INK 4;" OTRA VEZ (S/N
) "
220 LET A$=INKEY$
230 IF A$="S" THEN  GO TO 10
240 IF A$="N" THEN  GO TO 260
250 GO TO 220
260 INPUT 0: PRINT #0; FLASH 1; PAPER 6; INK 1; BRIGHT 1;"
      FIN DEL PROGRAMA        ": PAUSE 0
```

5. PASO : Lista de variables

A = Factor contaminación ambiental dependiente de la población
A$ = Variable de cadena (si,no)
B0 = Contaminación ambiental de partida
B1 = Contaminación ambiental en el siguiente período
I = Indice variable
P = Factor contaminación ambiental
R = Tasa de crecimiento anual
T = Tiempo
X0 = Población de partida
X1 = Población en el siguiente período
Z = Crecimiento de la población

6. PASO : Descripción del programa

Línea 10-60 : Título y aclaraciones

Línea 70-95 : Demanda de las informaciones INPUT

Línea 100-120 : Borrado de la pantalla e impresión de los encabezamientos de la tabla

Línea 130 : Definición de otros parámetros

Línea 140 : Cálculos para el siguiente período

Línea 150 : Más cálculos

Línea 155 : Si la población calculada es menor o igual a 0, imprimir mensaje y saltar a la línea 210

Línea 160-170 : Impresión de los resultados para los períodos

Línea 180 : Incremento del índice variable I

Línea 190 : Comprueba que no se impriman excesivos datos a la vez en pantalla.

Línea 200 : Si I es menor que T, X0 toma el valor de X1 y B0 el de B1, y el programa vuelve a la línea 140

Línea 210-250 : Consulta si se desean más calculos (en caso afirmativo volver a la línea 10 y en caso negativo final del programa.

Línea 260 : Impresión de FIN DEL PROGRAMA

7. PASO : Resultados

Si a la población de partida le damos p.e. el valor 1000, al horizonte de tiempo el valor 8 años, a la tasa de crecimiento anual el 5% y como factor de contaminación ambiental tomamos el valor 0.01, obtendremos:

AÑOS	POBLACION	%	CONT.
1	1050	5	15
2	945	-10	30.75
3	701.66	-25.75	44.93
4	421.49	-39.93	55.45
5	208.85	-50.45	61.77
6	90.29	-56.77	64.9
7	36.21	-59.9	66.25
8	14.03	-61.25	66.79

OTRA VEZ (S/N)

7

Capítulo 7 : Geografía/Historia

7.1 Consideración previa

En este capítulo pueden confeccionarse programas similares a los del capítulo "lenguas", es decir, ante todo programas de test y de consulta. También aquí nos limitaremos a utilizar ficheros de datos de muestra, explicando el montaje y funcionamiento del programa.

Para su utilización práctica, habrá que crear primero ficheros de datos completos y después introducirlos. Con este fin se completarán los DATAS, modificando el valor de N según el caso.

7.2 Fechas históricas

1. PASO : Presentación del problema

Presentamos un programa que puede utilizarse para aprender fechas históricas y para repasar estas fechas.

Con este fin, el ordenador le presenta al usuario acontecimientos históricos, requiriéndole para que introduzca la fecha correcta de estos acontecimientos históricos. El usuario puede decidir cuántas preguntas le debe formular el ordenador por cada serie de tests.

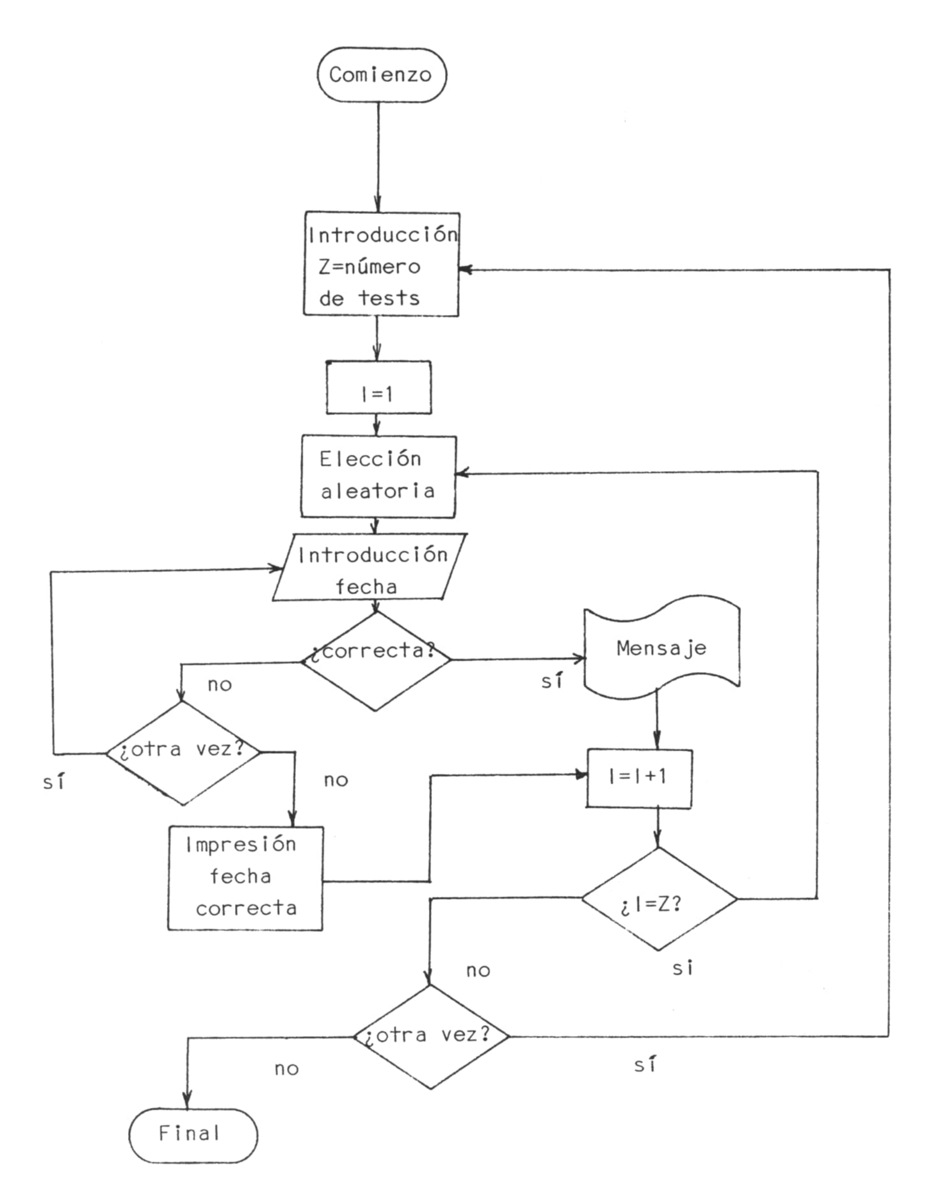
Si la respuesta del usuario es correcta, el ordenador le alaba; si, en cambio, la respuesta es incorrecta, el ordenador consulta si quiere probar otra vez. En caso de negativa por parte del usuario, el ordenador se ocupa de imprimir la respuesta correcta en pantalla.

2. PASO : Análisis del problema

La presentación del problema ya ha dado a entender claramente cómo debe procederse en este programa:

Una parte del programa está destinada a introducir información, también se debe preguntar al usuario cuántos tests desea hacer y el programa debe analizar cada respuesta. Por lo demás no se presentan dificultades especiales.

3. PASO : Diagrama de flujo



```
  10 REM G1-FECHAS HISTORICAS
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA REPASAR F
ECHAS HIS"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"TORICAS.
    "
  50 PRINT : PRINT "    ------------------------   "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------   "
  70 LET A$="Este programa utiliza como fichero 26 datos his
toricos seleccio-nados (contenidos en DATAS a partir de la l
inea 500 del programa). En caso de que se quisieran utilizar
 mas datos,deberan a/adirmas DATAS y cambiar el valor de N e
n la linea 160."
  80 PRINT
  90 FOR X=1 TO LEN A$
 100 BEEP .001,65: BEEP .04,30
 110 IF PEEK 23688=1 THEN  PRINT " "
 120 PRINT A$(X TO X);
 130 NEXT X
 140 PRINT #0; INK 2; BRIGHT 1; FLASH 1;" PARA PROSEGUIR PUL
SE UNA TECLA "
 145 LET a$=INKEY$
 146 IF a$="" THEN  GO TO 145
 150 CLS
 160 LET N=26
 165 PRINT AT 9,0; INK 3; BRIGHT 1;"------------------------
-------"
 170 PRINT AT 10,0; INK 3; BRIGHT 1;"|"; INK 7;" CUANTAS PRE
GUNTAS SE DESEAN "; INK 3; BRIGHT 1;"|"
 172 PRINT AT 11,0; INK 3; BRIGHT 1;"|";AT 11,30; INK 3; BRI
GHT 1;"|"
 175 PRINT AT 12,0; INK 3; BRIGHT 1;"|"; INK 7;"          REA
LIZAR ?            "; BRIGHT 1; INK 3;"|"
 180 PRINT AT 13,0; INK 3; BRIGHT 1;"-----------------------
--------"
 190 PRINT AT 18,0; FLASH 1; INK 6; BRIGHT 1;"  Por favor in
dicar un numero   "
 200 INPUT TAB 9;"numero = ? ";Z
 210 FOR I=1 TO Z
 220 CLS
 230 LET R=INT (RND*N+1)
 240 RESTORE
 250 FOR J=1 TO R
```

```
 260 READ j$: READ s$
 270 NEXT J
 280 PRINT AT 1,9;"CUANDO FUE : "
 290 PRINT AT 5,0;s$
 300 PRINT AT 10,8;"DIGA EL A/O :";: INPUT TAB 10;"A/O ? ";k
$: PRINT AT 10,21; FLASH 1;k$
 310 IF k$=j$ THEN  PRINT AT 17,12; FLASH 1; INK 2;"PERFECTO
": PAUSE 200: GO TO 380
 320 PRINT AT 15,2; FLASH 1; INK 6; PAPER 2; BRIGHT 1;"LA RE
SPUESTA NO ES CORRECTA"
 330 PRINT #0;TAB 6;"OTRO INTENTO (S/N)"
 340 LET A$=INKEY$
 350 IF A$="S" THEN  CLS : GO TO 280
 355 IF A$="N" THEN  GO TO 360
 356 GO TO 340
 360 PRINT AT 18,0;"LA FECHA HISTORICA CORRECTA ES : ";AT 20
,13; FLASH 1; BRIGHT 1; PAPER 2;j$: PAUSE 300
 380 NEXT I
 390 INPUT 0: PRINT #0;TAB 7;"UN TEST MAS (S/N)"
 400 LET A$=INKEY$
 410 IF A$="S" THEN  GO TO 150
 420 IF A$="N" THEN  GO TO 440
 430 GO TO 400
 440 PRINT #0; FLASH 1; PAPER 6; INK 4;"         FIN DEL PROG
RAMA         ": PAUSE 0
 500 DATA "1939","   EL PRINCIPIO DE LA SEGUNDA
                          GUERRA MUNDIAL"
 510 DATA "800","  LA CORONACION DEL EMPERADOR
                          CARLOMAGNO"
 520 DATA "1969","LA LLEGADA DEL HOMBRE A LA LUNA"
 530 DATA "1808"," LA GUERRA DE LA INDEPENDENCIA
                          ESPA/OLA"
 540 DATA "1492"," EL DESCUBRIMIENTO DE AMERICA"
 550 DATA "1936"," EL COMIENZO DE LA GUERRA CIVIL
                          ESPA/OLA"
 560 DATA "1871"," EL ESTABLECIMIENTO DEL IMPERIO
                          ALEMAN"
 570 DATA "1812","   LA LLEGADA DE NAPOLEON ANTE
                          MOSCU"
 580 DATA "1975","   LA MUERTE DEL GENERALISIMO
                          FRANCO"
 590 DATA "1914"," EL INICIO DE LA PRIMERA GUERRA
                          MUNDIAL"
 600 DATA "1469","     EL MATRIMONIO DE ISABEL
                          Y FERNANDO"
```

```
610 DATA "1494","   EL TRATADO DE TORDESILLAS"
620 DATA "1513"," EL DESCUBRIMIENTO DEL OCEANO
                             PACIFICO"
630 DATA "1519","   LA DISTINCION DE EMPERADOR
                            A CARLOS I"
640 DATA "1519","    EL DESEMBARCO DE CORTES
                            EN MEJICO"
650 DATA "1525","     LA BATALLA DE PAVIA"
660 DATA "1598","    LA MUERTE DE FELIPE II"
670 DATA "1558","     LA MUERTE DE CARLOS I"
680 DATA "1588","     EL FRACASO DE LA ARMADA
                            INVENCIBLE"
690 DATA "1618","   EL COMIENZO DE LA GUERRA DE
                           LOS 30 A/OS"
700 DATA "1648","      LA PAZ DE WESTFALIA"
710 DATA "1704","     LA OCUPACION INGLESA DE
                             GIBRALTAR"
720 DATA "1789","      LA REVOLUCION FRANCESA"
730 DATA "1939","    EL FIN DE LA GUERRA CIVIL
                             ESPA/OLA"
740 DATA "1776","LA INDEPENDENCIA DE LOS ESTADOS
                             UNIDOS"
750 DATA "1945","LA EXPLOSION DE LA PRIMERA BOMBA
                            ATOMICA"
```

Línea 10-146 : Título y aclaraciones

Línea 150-160 : N. líneas data y borrado de la pantalla

Línea 165-200 : Demanda del número de test

Línea 210-380 : Tests

--Línea 210-300 : Elección al azar de un acontecimiento histórico y demanda de la fecha como respuesta del usuario
--Línea 310-320 : Comprueba si la respuesta es o no correcta.
--Línea 330-356 : Consulta si se desea otro intento y según la respuesta procede en con secuencia
--Línea 360 : Si la respuesta ha sido ne gativa se imprime la respuesta correcta.
--Línea 380 : Vuelta a la línea 210.

Línea 390-430 : Consulta si se desea otra rueda de test; en caso afirmativo, borrado de la pantalla y vuelta a la línea 150. En caso negativo, final del programa.

Línea 440 : Impresión de FIN DEL PROGRAMA

Línea 500-750 : DATAS

7. PASO : Resultados

No es necesario ofrecer aquí los resultados. Nos parece más razonable que el lector pruebe el programa considerando los contenidos y los DATAS modificándolo después de acuerdo con sus propios deseos.

7.3 Las capitales de las Comunidades Autónomas

1. PASO : Presentación del problema

El programa que presentamos en este punto es también un llamado programa de entreno. El usuario debe responder a las preguntas que le formula el ordenador. Si el ordenador ofrece nombres de comunidades autónomas elegidos al azar, el usuario deberá introducir las capitales de estas comunidades; si en cambio el ordenador visualiza capitales, el usuario deberá citar las correspondientes comunidades.

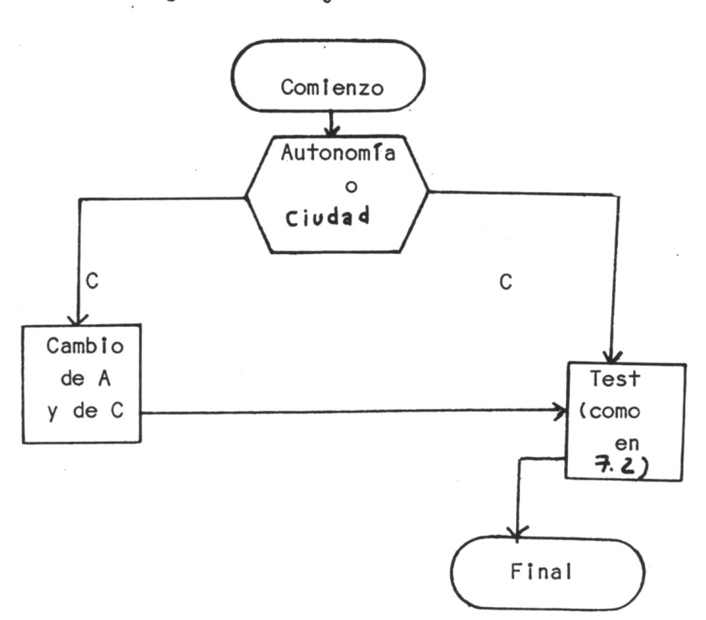
Llegados a este punto no hay problemas nuevos, de modo que podremos ser breves. Este programa tiene carácter ejemplar :

El usuario puede registrar fácilmente nuevos datos en los DATAS, obteniendo así un programa de consulta de aplicación general, que puede utilizarse para las más diversas asignaturas escolares.

2. PASO : Análisis del problema

Ya hemos mencionado que aquí no se presenta ningún problema adicional, de modo que podemos pasar rápidamente al siguiente paso. Todo lo referente a este programa ya ha sido suficientemente discutido en ejemplos precedentes.

3. PASO : Diagrama de flujo



```
  10 REM G2-COMUNIDADES AUTONOMAS
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA REPASAR
GEOGRAFIA"
  50 PRINT : PRINT "    ------------------------  "
  55 PRINT AT 3,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------  "
  70 LET A$="Este programa,  tras indicar unacomunidad auton
oma, nos preguntasu capital y al contrario. Puedeutilizarse
 tambien  para  otrosfines si  cambiamos los DATAS  apartir
de la linea 500, y el va-lor de la variable N en la linea160
."
  80 PRINT
  90 FOR X=1 TO LEN A$
 100 BEEP .001,65: BEEP .04,30
 110 IF PEEK 23688=1 THEN  PRINT " "
 120 PRINT A$(X TO X);
 130 NEXT X
 140 PRINT #0; INK 2; BRIGHT 1; FLASH 1;" PARA PROSEGUIR PUL
SE UNA TECLA "
 145 LET a$=INKEY$
 146 IF a$="" THEN  GO TO 145
 150 CLS
 160 LET N=11
 170 PRINT AT 6,0; PAPER 2; INK 4; BRIGHT 1; FLASH 1;"INDIQU
E AUTONOMIAS O CAPITALES"
 180 PRINT AT 14,3;"por favor indique A o C  : ": INPUT TAB
10;"? ";A$
 190 IF A$="C" THEN  LET C$="AUTONOMIA": GO TO 210
 200 IF A$="A" THEN  LET C$="CAPITAL": GO TO 210
 205 GO TO 180
 210 CLS : PRINT AT 9,3; FLASH 1; INK 3; PAPER 6; BRIGHT 1;"
 CUANTAS CONSULTAS DESEA :": INPUT TAB 4;"numero de consulta
s ? ";Z
 220 FOR I=1 TO Z
 230 CLS
 240 LET R=INT (RND*N+1)
 250 RESTORE
 260 FOR J=1 TO R
 270 READ L$: READ S$
 280 NEXT J
 290 IF A$="C" THEN  LET A$=L$: LET L$=S$: LET S$=A$
 300 PRINT AT 3,12;L$: PRINT AT 4,12;: FOR X=1 TO LEN (L$):
PRINT "-";: NEXT X
```

```
  10 REM E2-CRECIMIENTO LIMITADO
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA SIMULAR E
N UNA TA-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"BLA, UN PROCESO DE  CRE
CIMIENTO "
  45 PRINT  PAPER 4; INK 0; FLASH 1;"FRENADO.
        "
  50 PRINT : PRINT "    -------------------------   "
  55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    -------------------------  "
  70 PRINT AT 11,0;"POBLACION INICIAL......: ";: INPUT TAB 1
0;"? ";X0: PRINT X0
  80 PRINT AT 15,0;"TASA DE CRECIMIENTO  % : ";: INPUT TAB 1
0;"? ";R: PRINT R
  90 PRINT AT 19,0;"NUMERO DE A/OS.........: ";: INPUT TAB 1
0;"? ";T: PRINT T
 100 CLS
 110 PRINT AT 0,0; FLASH 1; PAPER 1; INK 6;"A/OS";AT 0,8;"PO
BLACION";AT 0,21;"CRECIMIENTO"
 120 PRINT AT 1,0;"----";AT 1,8;"---------";AT 1,21;"-------
----"
 130 LET I=1
 140 LET X1=X0+X0*(R/I)/100: LET X1=INT (X1*100+.5)/100
 150 LET Z=(X1-X0)/X0: LET Z=Z*100: LET Z=INT (Z*100+.5)/100
 160 PRINT
 170 PRINT TAB 0;I;TAB 10;X1;TAB 25;Z
 180 LET X0=X1: LET I=I+1
 190 IF I/10=INT (I/10) THEN  PRINT #0; FLASH 1; PAPER 4; IN
K 2;" PULSE UNA TECLA PARA CONTINUAR ": PAUSE 0: CLS
 200 IF I<=T THEN  GO TO 140
 210 PRINT #0;TAB 8; FLASH 1; PAPER 6; INK 4;" OTRA VEZ (S/N
) "
 220 LET A$=INKEY$
 230 IF A$="S" THEN  GO TO 10
 240 IF A$="N" THEN  GO TO 260
 250 GO TO 220
 260 INPUT 0: PRINT #0; FLASH 1; PAPER 6; INK 1; BRIGHT 1;"
      FIN DEL PROGRAMA         ": PAUSE 0
```

5. PASO : Lista de variables

A$ = Variable de cadena
C$ = Variable auxiliar que según se elija toma el valor "capital" o ''autonomía''
H$ = Variable auxiliar para el cambio
I = Indice variable
J = Indice variable
L$ = Autonomías
N = Número de registros
S$ = Capitales
X$ = Introducción hecha por el usuario

6. PASO : Descripción del programa

Línea 10-130 : Título y aclaraciones

Línea 140-146 : Espera

Línea 150 : Borrado de la pantalla

Línea 160 : Se asigna a la variable N el valor 11

Línea 170-205 : Consulta si deben visualizarse autonomías o capitales

Línea 190 : Si se eligen capitales, la variable auxiliar C$ pasa a ser ''autonomía''

Línea 200 : Si se eligen autonomías, la variable auxiliar C$ pasa a ser ''capital''

Línea 210 : Demanda del número de consultas

Línea 220-410 : Formulación del número de consultas elegido

240 : Elección al azar de un par de valores
250 : Restauración del fichero de datos
290 : Si se ha seleccionado C en el menú (ver 190-200), se intercambian S$ y L$
300 : Impresión de la pregunta
320 : Introducción de la respuesta por parte del usuario
330 : En caso correcto, seguir en 410
400 : Impresión de la respuesta correcta

Línea 420-460 : Consulta si se desea efectuar un test más.

Línea 470 : Fin del programa.

Línea 500-580 : DATAS

7. PASO : Resultados

Tampoco en este programa resulta necesario ofrecer los resultados. El análisis del programa, o aún mejor, la ejecución del mismo muestra claramente lo que ocurre en cada momento.

7.4 El desarrollo de la población en diferentes países

1. PASO : Presentación del problema

Presentamos aquí un programa que pronostica el desarrollo de la población de diferentes países del mundo hasta el año 2000.

Los países o grupos de países seleccionados para el estudio son:

1.- Mundo (comprendiendo a todos los países)
Países industrializados
Países en vías de desarrollo

2.- RP China
India
Japón
USA
URSS

3.- República Federal de Alemania
República Democrática Alemana

2. PASO : Análisis del problema

Los pronósticos de población planteados requieren el decidirse por un determinado método de pronosticación.

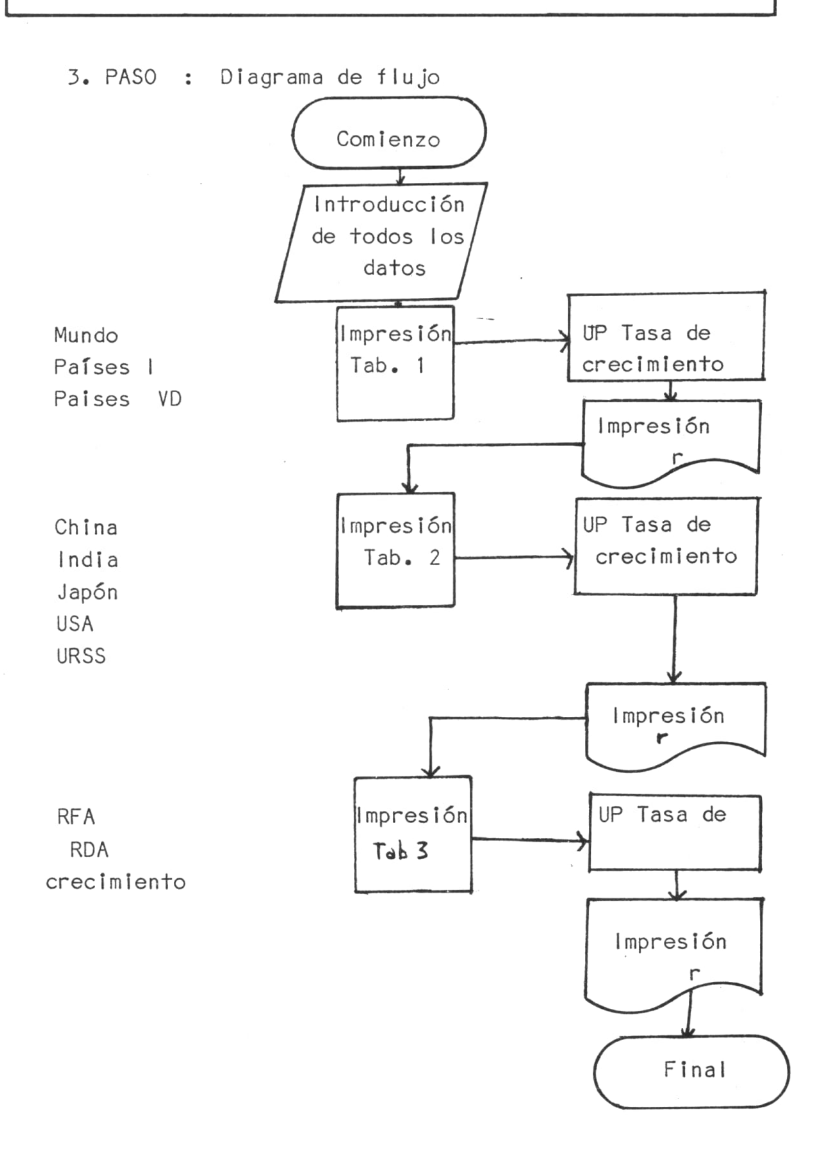
Existe un gran número de métodos distintos, que aquí no pueden ser discutidos uno a uno.

Para este programa hemos extraído los valores obtenidos en pronósticos modelo como los publicados en el informe GLOBAL 2000 (Washington 1980).

A partir de estos datos calculamos además las tasas de crecimiento medias anuales, para poder ofrecer a través del programa una información adicional.

Desde el punto de vista de la programación no aparecen dificultades importantes, si exceptuamos que, debido al volumen de resultados, en diversos puntos del programa es necesario efectuar interrupciones.

3. PASO : Diagrama de flujo



```
  10 REM G-3 TABLA DE POBLACION
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA REPRESENT
AR EN UNA"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"TABLA EL DESARROLLO DE
LA POBLA-"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"CION DE DIFERENTES PAIS
ES.      "
  50 PRINT : PRINT "    -----------------------  "
  55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    -----------------------  "
  70 GO SUB 720
  80 CLS
  90 DIM B(10,6): DIM J(6): DIM W(10)
 100 FOR I=1 TO 6
 110 READ X
 120 LET J(I)=X
 130 NEXT I
 140 FOR L=1 TO 10
 150 FOR I=1 TO 6
 160 READ X
 170 LET B(L,I)=X
 180 NEXT I
 190 NEXT L
 200 LET A$="MUNDO": LET B$="IND."
 210 LET C$="V.D.": LET D$="CHINA"
 220 LET E$="INDIA": LET F$="JAPON"
 230 LET G$="USA": LET H$="URSS"
 240 LET I$="RFA": LET J$="RDA"
 250 PRINT AT 2,0;"   "; PAPER 2;"A/O"; PAPER 0;"   "; PAPER
 2;A$; PAPER 0;"   "; PAPER 2;B$; PAPER 0;"   "; PAPER 2;C$
 260 PRINT
 270 GO SUB 810
 280 PRINT AT 4,0
 290 FOR I=1 TO 6
 300 PRINT : PRINT "   ";J(I);"   ";B(1,I);"   ";B(2,I);"
";B(3,I)
 310 NEXT I
 320 PRINT
 330 GO SUB 810
 340 FOR L=1 TO 3
 350 LET X=B(L,1): LET Y=B(L,6)
 360 GO SUB 760
 370 NEXT L
```

```
380 PRINT AT 20,0; INK 4;" TASA %";"   ";W(1);"   ";W(2);"
  ";W(3)
390 GO SUB 720
400 CLS
410 PRINT  PAPER 2;"A/O"; PAPER 0;"  "; PAPER 2;D$; PAPER 0
;" "; PAPER 2;E$; PAPER 0;" "; PAPER 2;F$; PAPER 0;"  "; PAP
ER 2;G$; PAPER 0;" "; PAPER 2;H$
420 PRINT
430 GO SUB 810: PRINT
440 LET T=0
450 FOR I=1 TO 6
460 PRINT AT 3+I+T,0;J(I);AT I+3+T,6;B(4,I);AT I+3+T,12;B(5
,I);AT I+3+T,18;B(6,I);AT 3+T+I,24;B(7,I);AT 3+I+T,29;B(8,I)
470 LET T=T+1
480 NEXT I
490 PRINT : GO SUB 810
500 FOR L=4 TO 8
510 LET X=B(L,1): LET Y=B(L,6)
520 GO SUB 760
530 NEXT L
540 PRINT : PRINT  INK 4;" % ";"   ";W(4);"  ";W(5);"  ";W(
6);" ";W(7);" ";W(8)
550 GO SUB 720: CLS
560 PRINT "     "; PAPER 2;"A/O"; PAPER 0;"       "; PAPER 2
;I$; PAPER 0;"     "; PAPER 2;J$
570 PRINT : GO SUB 810
580 LET T=0
590 FOR I=1 TO 6
600 PRINT AT 3+I+T,4;J(I);AT 3+I+T,14;B(9,I);AT 3+I+T,22;B(
10,I)
610 LET T=T+1
620 NEXT I
630 PRINT : GO SUB 810
640 FOR L=9 TO 10
650 LET X=B(L,1): LET Y=B(L,6)
660 GO SUB 760
670 NEXT L
680 PRINT
690 PRINT  INK 4;"     %  ";"      ";W(9);"     ";W(10)
700 GO SUB 720
710 CLS : PRINT AT 12,8; FLASH 1; PAPER 6; INK 3; BRIGHT 1;
"FIN DEL PROGRAMA": PAUSE 0: STOP
720 REM LECTURA DEL TECLADO
730 PRINT #0;TAB 8; FLASH 1; INK 5;"PULSE UNA TECLA"
740 LET Q$=INKEY$
```

```
750 IF Q$="" THEN  GO TO 740
755 RETURN
760 REM TASA DE CRECIMIENTO
770 LET R=EXP (LN (Y/X)/25)-1
780 LET R=R*100: LET R=INT (R*100+.5)/100
790 LET W(L)=R
800 RETURN
810 REM RAYADO
820 FOR I=0 TO 31
830 PRINT "-";
840 NEXT I
850 PRINT
860 RETURN
870 DATA 1975,1980,1985,1990,1995,2000
880 REM           MUNDO
890 DATA 4134,4549,5013,5545,6143,6798
900 REM PAISES INDUSTRIALIZADOS
910 DATA 1131,1174,1224,1276,1327,1377
920 DATA 3003,3375,3789,4269,4816,5420
930 REM REPUBLICA POPULAR CHINA
940 DATA 978,1071,1151,1241,1348,1468
950 REM         INDIA
960 DATA 618,694,786,894,1013,1142
970 REM         JAPON
980 DATA 112,117,122,127,131,135
990 REM   ESTADOS UNIDOS
1000 DATA 214,222,235,248,260,270
1010 REM  UNION SOVIETICA
1020 DATA 254,268,282,296,310,323
1030 REM REPUBLICA FEDERAL ALEMANA
1040 DATA 61.8,61.7,60,58.6,57.8,56.2
1050 REM REPUBLICA DEMOCRATICA ALEMANA
1060 DATA 16.8,16.7,16.6,16.4,16.2,16.1
```

5. PASO : Lista de variables

B = Población
I = Indice variable
J = Año
L = Indice variable
L$ = Países
WR = Tasa de crecimiento
X = Valor de partida (1975)
Y = Valor final (2000)

6. PASO : Descripción del programa

Línea 10-70 : Título, comentarios del programa y salto a la subrutina 720.

Línea 80-240 : Lectura de todos los datos de partida

Línea 250 : Encabezamiento de la tabla

Línea 270 : Salto a la subrutina de subrayado

Línea 280-330 : Impresión de los valores de la tabla

Línea 340-370 : Determinación de las tasas de cre cimiento medias anuales para los paí ses L=1, L=2 y L=3

Línea 350 : Toma del valor inicial y del valor final
Línea 360 : Salto a la subrutina de cálculo de la tasa de crecimiento

Línea 380 : Impresión de tasas de crecimiento

Línea 390-400 : Salto a subrutina de espera y borrado

Línea 410-540 : Mismo procedimiento que en la línea 270-410, pero para la segunda tabla

Línea 550-700 : Lo mismo para la tercera tabla

Línea 710 : Final del programa

Línea 720-755 : Subrutina de espera

Línea 760-800 : Subrutina Tasa de crecimiento media anual (se calcula con ayuda de logaritmos, ver línea 770)

Línea 810-860 : Subrutina Raya :
Mediante encadenamiento de signos menos se dibuja una raya en la pantalla

Línea 870-1060 : DATAS

7. PASO : Resultados

Este programa genera en la pantalla tres tablas consecutivas

1. Tabla (referencias en millones)

AÑO	MUNDO	PAISES I	PAISES VD
1975	4134	1131	3003
1980	4549	1174	3375
1985	5013	1224	3789
1990	5545	1276	4269
1995	6143	1327	4816
2000	6798	1377	5420
TASA %	2.01	.79	2.39

2. Tabla (referencias en millones)

AÑO	CHINA	INDIA	JAPON	USA	USSR
1975	978	618	112	214	254
1980	1071	694	117	222	268
1985	1151	786	122	235	282
1990	1241	894	127	248	296
1995	1348	1013	131	260	310
2000	1468	1142	135	270	323
%	1.64	2.49	.75	.93	.97

3. Tabla (referencias en millones)

AÑO	RFA	RDA
1975	61.8	16.8
1980	61.7	16.7
1985	60	16.6
1990	58.6	16.4
1995	57.8	16.2
2000	56.2	16.1
%	-.38	-.17

FIN DEL PROGRAMA

8

Capítulo 8 : La economía

8.1 Consideración previa

Durante mucho tiempo, el campo de aplicación fundamental de los ordenadores fue el de la Economía, y en especial el empresarial y comercial. Puede decirse que hoy en día sigue siendo en estos campos, donde se emplean con más frecuencia los ordenadores.

Los problemas tratados en estos campos se prestan a ser resueltos por medio de ordenadores, hoy en día también con microordenadores.

Algunos de los problemas fundamentales de la Economía, por lo menos en lo referente a los cálculos, son tratados también en las escuelas; en escuelas empresariales de forma más intensiva, pero también en otros tipos de escuelas, donde tampoco se evitan este tipo de problemas.

También aquí, los problemas seleccionados sólo son representativos de otros muchos, que por motivos de espacio no pueden ser incluídos en esta obra.

8.2 El cálculo de intereses

1. PASO : Presentación del problema

Imaginemos que alguien lleva un determinado importe al banco, del que se deducirá el pago de un determinado interés por la imposición del mismo. Este interés puede ser fijado arbitrariamente.

Al final de cada año aumenta el capital en un determinado tanto por ciento. Hay que tener en cuenta, además, que a partir del segundo año se obtiene un interés sobre el capital más el interés del año anterior (interés compuesto).

Vamos a desarrollar ahora un programa que calculará - para un importe inicial arbitrario y para un rédito cualquiera- el importe final obtenido después de una cifra determinada de años.

Una programa como este puede servir además como patrón para la simulación de diversos procesos de crecimiento.

2. PASO : Análisis del problema

Si llamamos al capital de partida K0, al rédito P (%) y al tiempo que se tiene el dinero colocado en el banco T (años).

Después del primer año, el capital resultante X1 es :

$$X1 = X0 + X0 * P/100 = X0 * (1 + P/100)$$

Tras el segundo año se obtiene del mismo modo:

$$X2 = X1 + X1 * P/100 = X1 * (1 + P/100)$$

$$= X0 * (1 + P/100) * (1 + P/100)$$

$$= X0 * (1 + P/100)^2$$

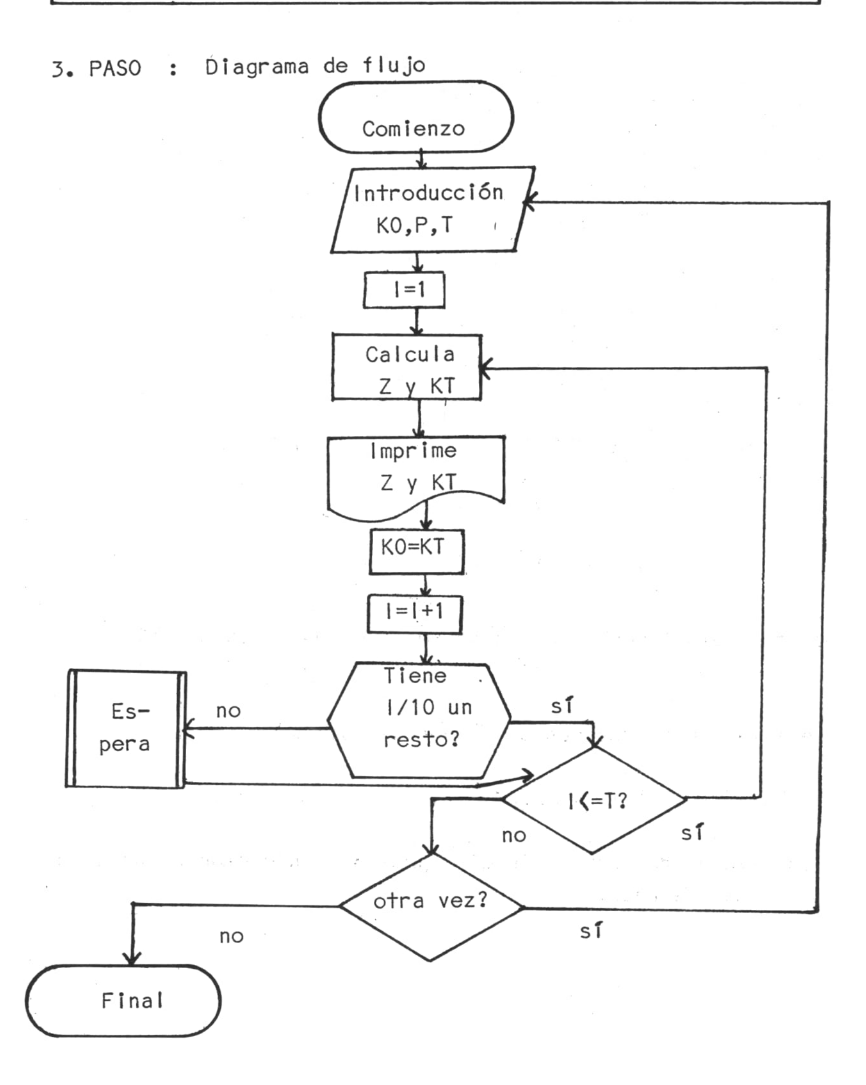
(Estas relaciones ya las conocemos de capítulos anteriores).

De esta forma, después de T años se obtiene:

$$X^T = X0 * (1 + P/100)^T$$

Con ayuda de esta fórmula podemos confeccionar ahora un programa adecuado.

3. PASO : Diagrama de flujo



```
  10 REM E-1 CALCULO DE INTERESES
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA EL CALCUL
O DE INTE"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"RESES.
      "
  50 PRINT : PRINT "     -----------------------   "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "     -----------------------   "
  70 PRINT AT 10,3;"CAPITAL...............= ";: INPUT TAB 10
;"Capital ?  ";K0: PRINT K0
  80 PRINT AT 14,3;"REDITO ANUAL.........= ";: INPUT TAB 10
;"Redito ?  ";P: PRINT P
  90 PRINT AT 18,3;"TIEMPO DE IMPOSICION..= ";: INPUT TAB 10
;"Tiempo ?  ";T: PRINT T
 100 PAUSE 50: CLS
 110 PRINT AT 2,1; PAPER 2;"A/O"; PAPER 0;"      "; PAPER 2;"
INTERESES"; PAPER 0;"      "; PAPER 2;"CAPITAL"
 120 PRINT  INK 4;" ---     ---------     -------"
 130 LET M=6
 140 FOR I=1 TO T
 150 LET Z=K0*P/100
 160 LET KT=K0+Z
 170 LET Z=INT (Z*100+.5)/100
 180 LET KT=INT (KT*100+.5)/100
 190 PRINT AT M,2;I;AT M,12;Z;AT M,24;KT
 200 LET K0=KT
 210 IF I/7=INT (I/7) THEN  GO SUB 290
 220 LET M=M+2: IF M>18 THEN  LET M=6
 230 NEXT I
 240 INPUT 0: PRINT #0;TAB 7; FLASH 1; INK 3; PAPER 5;"OTRO
CALCULO (S/N)"
 250 LET A$=INKEY$
 260 IF A$="S" THEN  CLS : GO TO 10
 270 IF A$="N" THEN  GO TO 280
 275 GO TO 250
 280 CLS : PRINT AT 11,6; BRIGHT 1; PAPER 4; INK 2;"FIN DEL
PROGRAMA": PAUSE 0: STOP
 290 REM SUBRUTINA DE ESPERA
 300 PRINT #0; FLASH 1; INK 2; BRIGHT 1;" Pulse una tecla pa
ra continuar"
 310 LET A$=INKEY$
 320 IF A$="" THEN  GO TO 310
 330 CLS : PRINT AT 2,1; PAPER 2;"A/O"; PAPER 0;"      "; PAP
ER 2;"INTERESES"; PAPER 0;"      "; PAPER 2;"CAPITAL"
 340 PRINT  INK 4;" ---     ---------     -------"
 350 RETURN
```

5. PASO : Lista de variables

A$ = Variable de cadena para las respuestas
I = Indice variable
K0 = Capital inicial
KT = Capital en los períodos siguientes
P = Rédito anual en tanto por ciento
T = Tiempo de imposición en años
Z = Interés anual

6. PASO : Descripción del programa

Línea 10-60 : Título

Línea 70-90 : Demanda de los datos de partida

Línea 100-120 : Limpieza de la pantalla e impresión de un encabezamiento de tabla

Línea 130-230 : Cálculos

150 : Interés
160 : Capital tras el siguiente período
170-180 : Redondeo
190 : Impresión
200 : K0 toma el valor KT, para poder entrar en el siguiente cálculo
210 : Después de cada 7 períodos interrupción mediante el salto a la subrutina 290

Línea 240-275 : Final del programa, a menos que se desee seguir calculando (vuelta a la línea 10)

Línea 290-350 : Subrutina para esperar cuando la pantalla está llena:

300 : Indicación
310-320 : La variable A$ toma el valor de la tecla que se pulsa. Si no pulsa ninguna vuelta a 310.
330-340 : Borrado de la pantalla y nueva impresión del encabezamiento
350 : Salto de vuelta

7. PASO : Resultados

Si introducimos el capital inicial Ptas. 100, el rédito 5.5% y el tiempo 6 años, se obtiene :

AÑO	INTERES	CAPITAL
1	5.5	105.5
2	5.2	113.3
3	6.12	117.42
4	6.46	123.88
5	6.81	130.69
6	7.19	137.88

OTRO CALCULO (S/N)

8.3 Amortización de una hipoteca

1. PASO : Presentación del problema

De entre los problemas de la Economía, aquéllos que están relacionados con el pago de créditos e hipotecas juegan también un papel importante. Hay programas de ordenador que pueden calcular el tiempo de amortización bajo diversas condiciones de partida, como p.e. valor de la deuda, el interés o las modalidades de pago.

El programa siguiente calcula el tiempo de amortización para diferentes importes e intereses o cuando se han acordado cuotas de amortización de distintas cuantías.

Puede caerse fácilmente en la cuenta de que este programa es aplicable de forma más general en problemas de este tipo.

2. PASO : Análisis del problema

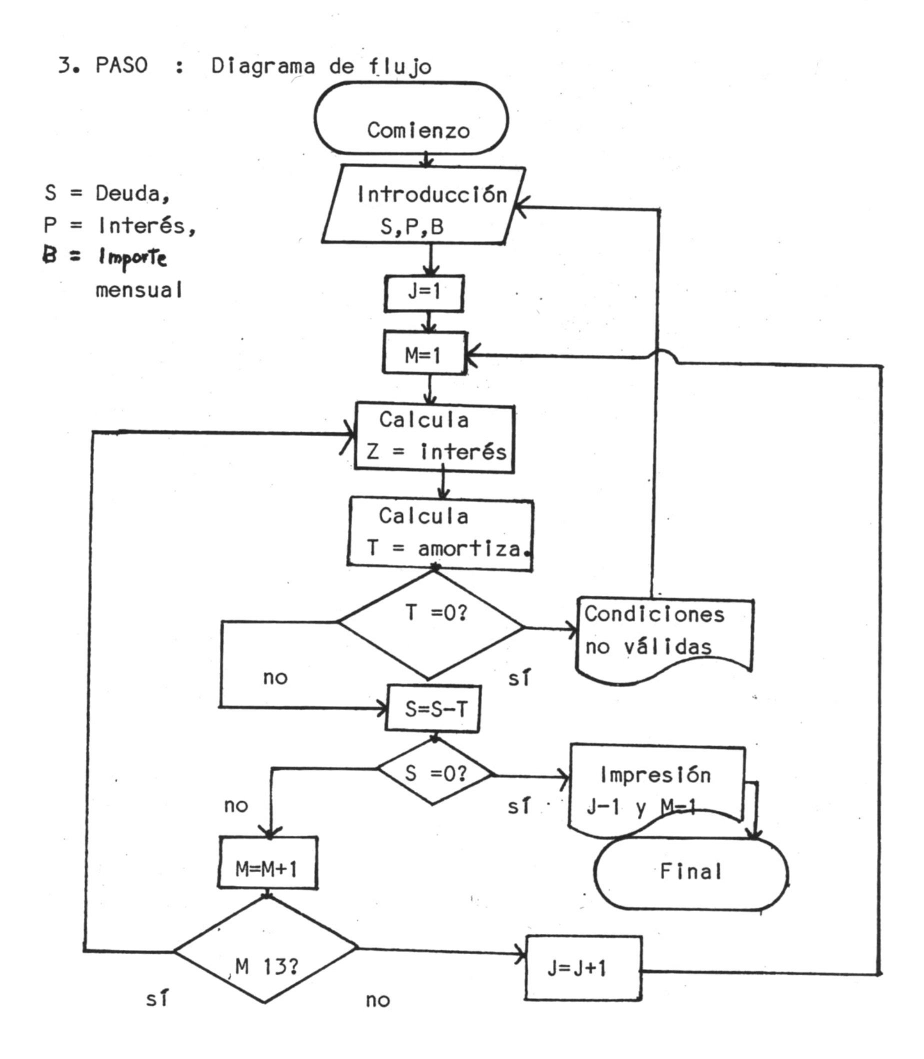
Para explicar la resolución del problema nos basaremos en el siguiente ejemplo:

Alguien solicita una hipoteca de S Ptas. y acuerda con el banco concesor un pago mensual de B Ptas. De este importe mensual B deben restarse primero los intereses por la deuda. El importe que queda después de restar los intereses puede aplicarse para saldar la deuda - suponiendo que quede un resto positivo.

El programa que vamos a confeccionar debe calcular mes a mes los intereses y el capital de amortización. Si este importe es igual o menor a 0, el programa debe imprimir un mensaje para avisar de que bajo esas condiciones ya no es posible amortizar la deuda.

Debe imprimirse año por año un estado de la deuda, y cuando la deuda sea cero o negativa, deberá indicarse la duración del proceso de amortización.

3. PASO : Diagrama de flujo



```
  10 REM E-2 CALCULO DE HIPOTECAS
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA CALCULAR
EL TIEMPO"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"DE AMORTIZACION DE UNA
HIPOTECA."
  50 PRINT : PRINT "    -----------------------  "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    -----------------------  "
  70 PRINT AT 10,3;"CUOTA ANUAL...........= ";: INPUT TAB 10
;"Cuota ?  ";B: PRINT B
  80 PRINT AT 14,3;"INTERES EN %..........= ";: INPUT TAB 10
;"Interes ?  ";P: PRINT P
  90 PRINT AT 18,3;"DEUDA HIPOT. ACTUAL...= ";: INPUT TAB 10
;"Deuda ?  ";S: PRINT S
 100 PAUSE 50: CLS
 110 PRINT AT 2,6; PAPER 2;"A/O"; PAPER 0;"            "; PAP
ER 2;"DEUDA"
 120 PRINT  INK 4;"      ---            -----"
 130 LET H=5
 140 LET J=1
 150 LET M=1
 160 LET Z=((P/100)*S)/12: LET T=B-Z
 170 IF T<=0 THEN  GO TO 230
 180 LET S=INT ((S-T)*100+.5)/100
 190 IF S<=0 THEN  GO TO 290
 200 LET M=M+1
 205 IF M<13 THEN  GO TO 160
 210 PRINT AT H,7;J;AT H,19;S
 220 LET J=J+1: LET H=H+2: GO TO 150
 230 CLS : PRINT AT 7,0;"LAS CONDICIONES NO PERMITEN AMOR"
 240 PRINT : PRINT "TIZAR LA DEUDA.  INTRODUZCA NUE-"
 250 PRINT : PRINT "VOS DATOS POR FAVOR."
 260 PRINT #0;TAB 9;"Pulse una tecla"
 270 IF INKEY$="" THEN  GO TO 270
 280 CLS : GO TO 10
 290 INPUT 0: PRINT #0;TAB 9; PAPER 6; INK 3;"PULSE UNA TECL
A"
 300 IF INKEY$="" THEN  GO TO 300
 310 CLS : PRINT AT 10,7;"TIEMPO DE AMORTIZACION : "
 320 PRINT AT 13,10; BRIGHT 1; INK 4;J-1; INK 7;" A/OS y ";
INK 4;M-1; INK 7;" MESES"
 330 INPUT 0: PRINT #0;TAB 9; PAPER 2;"OTRO CALCULO (S/N)"
 340 LET A$=INKEY$
 350 IF A$="S" THEN  CLS : GO TO 10
 360 IF A$="N" THEN  GO TO 370
 365 GO TO 340
 370 CLS : PRINT AT 11,9;"FIN DEL PROGRAMA": PAUSE 0: STOP
```

5. PASO : Lista de variables

A$ = Variable de cadena (si,no)
B = Cuota mensual
I = Indice variable
J = Contador de años
M = Contador de meses
P = Interés anual de la deuda
T = Importe mensual de amortización
Z = Pago mensual de intereses

6. PASO : Descripción del programa

Línea 10-60 : Título

Línea 70-90 : Petición de las informaciones de partida

Línea 100-120 : Impresión del encabezamiento de la tabla

Línea 130-150 : Fijación del contador de años (J) y del contador de meses M

Línea 160-180 : Cálculo del interés mensual, de la cuota mensual y de la nueva deuda

Línea 170 : En caso de no poder amortizar, seguir en 230

Línea 190 : Una vez saldada la deuda, seguir en 290

Línea 200-205 : Mes siguiente. Si M es menor que 13 seguir en 160.

Línea 210 : Impresión de la deuda actual

Línea 220 : Año siguiente y seguir en línea 150

Línea 230-280 : Mensaje sobre condiciones no permitidas y vuelta a la línea 10

Línea 290-320 : Impresión del tiempo de amortización en años y en meses (piensa porque se le resta cada vez el valor 1 a J y a N)

Línea 330-370 : Final del programa, a menos que no se desee un nuevo cálculo (entonces retorno a la línea 10 tras borrar la pantalla)

7. PASO : Resultados

Introduzcamos estos datos cuando el ordenador nos lo pida:

```
CUOTA ANUAL           : ? 500
INTERES EN %          : ? 6.5
DEUDA HIPOT. ACTUAL   : ? 40000
```

y obtendremos en el transcurso del programa :

```
     AÑO        DEUDA
--------------------------
      1        36496.86
      2        32759.09
      3        28771.02
      4        24515.85
      5        19975.7
      6        15131.48
      7        9962.85
      8        4448.07

TIEMPO DE AMORTIZACION :

  8 AÑOS Y 9 MESES.

 OTRO CALCULO  (S/N) ?
```

8.4 Media aritmética

1. PASO : Presentación del problema

En algunos problemas de economía resulta necesario calcular el promedio de un número determinado de datos; normalmente se trata de la media aritmética.

El programa siguiente efectúa este cálculo para un número cualquiera de datos.

2. PASO : Análisis del problema

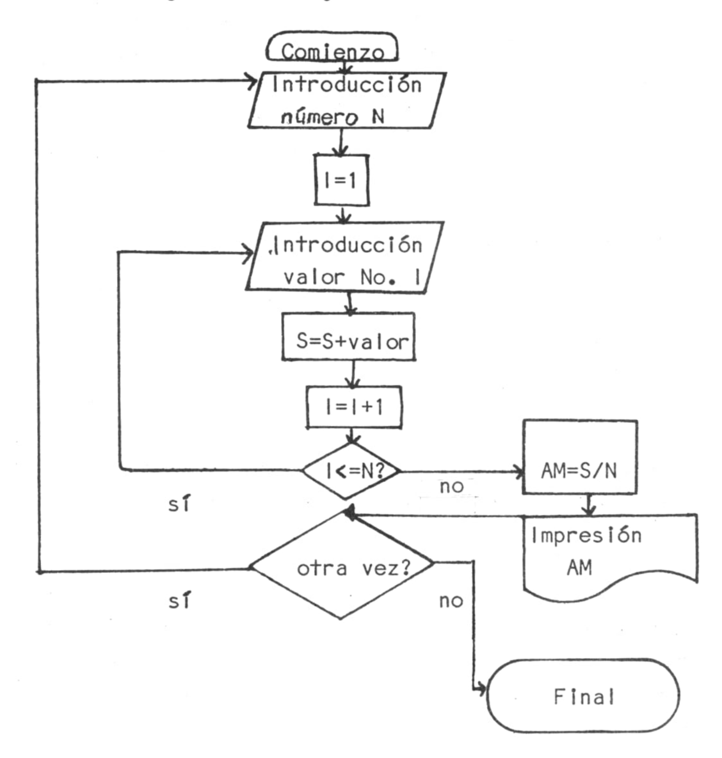
El análisis del problema no presenta especiales dificultades.

Primero hay que procurar que se introduzca un número no fijado de datos, por lo que se ha de informar al ordenador de cuántos valores se van a introducir.

Al introducir los valores puede efectuarse automáticamente la suma acumulativa de los mismos (la media aritmética no es más que la suma de todos los valores dividida entre el número de ellos). La división de la suma resultante del número de valores puede imprimirse como resultado del programa.

Para poder efectuar más cálculos, el programa se bifurca al principio. Si no se desea continuar finaliza él mismo.

3. PASO : Diagrama de flujo



4. PASO : Programa

```
  10 REM E-3   VALOR MEDIO
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA CALCULAR
UNA MEDIA"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"ARITMETICA.
      "
  50 PRINT : PRINT "    -----------------------  "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    -----------------------  "
  65 LET S=0
  70 PRINT AT 12,3;"CUANTOS VALORES......: ";: INPUT TAB 12;
"? ";N: PRINT N
  75 IF N=0 THEN  GO TO 180
  80 FOR I=1 TO N
  90 PRINT AT 17,3;I;" VALOR..............:": INPUT TAB 12;"
? ";X
 100 LET S=S+X
 110 NEXT I
 120 LET AM=S/N
 130 PRINT AT 17,3; INK 4; BRIGHT 1;"MEDIA ARITMETICA.....=
";AM
 140 PRINT #0;TAB 7; PAPER 2; INK 4; FLASH 1;"OTRO CALCULO (
S/N)"
 150 LET A$=INKEY$
 160 IF A$="S" THEN  CLS : GO TO 10
 170 IF A$="N" THEN  GO TO 180
 175 GO TO 150
 180 CLS : PRINT AT 11,8; FLASH 1; PAPER 5; INK 6;"FIN DEL P
ROGRAMA": PAUSE 0
```

5 . PASO : Lista de variables

AM = Media aritmética
A$ = Variable de cadena (si/no)
I = Indice variable
N = Número de valores a promediar
S = Suma de los valores a promediar
X = Valor que entra en el cálculo del valor medio

6. PASO : Descripción del programa

Línea 10-60 : Título

Línea 70 : Indicación del número de valores a promediar

Línea 80-110 : Introducción de un valor tras otro y suma acumulativa de todos los valores

Línea 120-130 : Cálculo de la media aritmética e impresión del resultado

Línea 140-180 : Final del programa, a menos que no se desee continuar calculando

7. PASO : Resultados

Si introducimos p.e., que existen tres valores, el programa pedirá exactamente tres valores.

Si éstos son p.e. los valores 5, 7 y 9, obtendremos:

MEDIA ARITMETICA = 7

OTRO CALCULO (S/N) ?

8.5 Distribución de frecuencias

1. PASO : Presentación del problema

A la hora de valorar volúmenes de datos grandes, que es cuando los ordenadores se muestran especialmente útiles, uno de los primeros pasos de la investigación estadística consiste en realizar una llamada distribución de frecuencias.
Se trata de asignar mediante una tabla o un gráfico la frecuencia con que aparecen los valores característicos (o a las clases de valores característicos) de la variable objeto de la investigación.

El programa siguiente realiza esta distribución de frecuencias tanto gráficamente como en forma de tabla. Como material de valoración utilizaremos la distribución por edades de la población de la RFA en el año 1980.

Esta distribución por edades de la población federal alemana es la siguiente:

Grupos de edades	Porcentaje
menos de 10	10.3
10 hasta 20	16.4
20 hasta 30	14.6
30 hasta 40	13.4
40 hasta 50	14.2
50 hasta 60	11.9
60 hasta 70	8.9
70 hasta 80	7.8
80 hasta 90	2.4
90 y más	0.2

2. PASO : Análisis del problema

Para el análisis del problema partiremos de la base de que disponemos de los datos tal como aparecen en la tabla, de forma que puedan ser utilizados como material de partida para el programa a confeccionar.

De no ser este el caso, y tuviéramos datos sueltos obtenidos en una encuesta, deberíamos procurar en una primera parte del programa la clasificación de estos datos sueltos por grupos de edades (esto puede conseguirse mediante una serie de instrucciones IF...THEN, que comprobarían cada uno de los valores de edad).

Una vez hecho esto, podemos relativizar los valores obtenidos para cada grupo de edades, comparándolos con la totalidad. De este modo, obtenemos los porcentajes que ya aparecen en la tabla anterior.

Esta parte del programa podría resumirse esquemáticamente de la forma siguiente:

```
100 INPUT "Edad siguiente : ";X
110 IF X<10 THEN LET Z1=Z1+1 : GOTO 100
120 IF X<20 THEN LET Z2=Z2+1 : GOTO 100
...
etc.
200 LET R1=(Z1/N)*100
210 LET R2=(Z2/N)*100
...
etc.
```

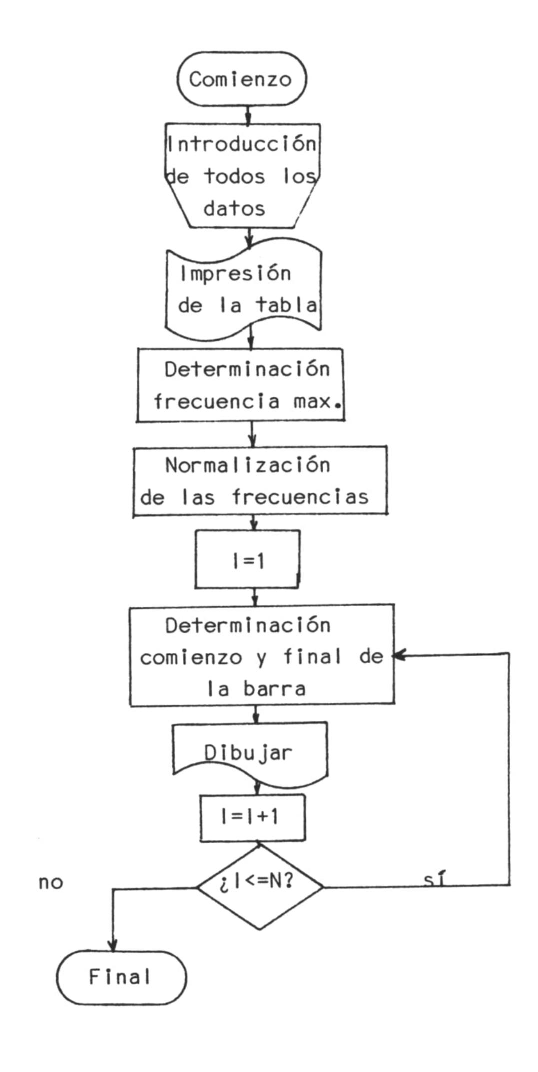
En este ejemplo se trata únicamente de representar en forma de tabla los valores introducidos, cosa que lógicamente no ofrece ninguna dificultad desde el punto de vista de la programación.

La representación gráfica de esta distribución de frecuencias, en cambio, es más complicada.

Para poder aprovechar óptimamente la pantalla en la representación gráfica, determinaremos, en primer lugar,cuál es la frecuencia mayor y, a continuación, convertiremos todas las demás frecuencias de forma que la primera ocupe 15 líneas en vertical de la pantalla.

Las frecuencias en sí se representan en forma de barras verticales. Para que el tamaño de estas barras sea correcto, debe determinarse su correspondiente punto final. Junto con el punto de origen obtendremos dos posiciones de pantalla por barra y ya tan solo será necesario conectar con una línea.

3. PASO : Diagrama de flujo



```
  10 REM E-4   DISTRIBUCION
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA REPRESENT
AR EN UNA"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"TABLA Y GRAFICAMENTE UN
A DISTRI-"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"BUCION ESTADISTICA.
        "
  50 PRINT : PRINT "    ------------------------   "
  55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------   "
  70 PRINT AT 10,0;"Se representa la distribucion de"
  80 PRINT AT 13,0;"edades en la  Republica  Federal"
  90 PRINT AT 16,0;"Alemana en 1980."
 100 PRINT #0;TAB 8; FLASH 1; PAPER 2;"PULSE UNA TECLA"
 110 IF INKEY$="" THEN  GO TO 110
 120 CLS
 125 DIM F(10): DIM G(10)
 130 FOR I=1 TO 10
 140 READ X
 150 LET F(I)=X
 160 NEXT I
 170 PRINT "         "; INK 3; BRIGHT 1;"EDAD";"          ";
INK 3; BRIGHT 1;"PORCENTAJE"
 180 FOR I=0 TO 31: PRINT AT 1,I; INK 6; BRIGHT 1;"-";: NEXT
 I
 190 LET H=1
 200 FOR I=1 TO 10
 210 LET A=(I-1)*10
 220 LET B=I*10
 230 PRINT AT 2+H,0;A;AT 2+H,3;"HASTA MENOS DE";AT 2+H,18;B;
AT 2+H,23;F(I);AT 2+H,30;"%"
 240 LET H=H+2
```

```
250 NEXT I
260 PRINT #0;TAB 8; PAPER 2; FLASH 1;"PULSE UNA TECLA"
270 IF INKEY$="" THEN  GO TO 270
280 LET FM=0: CLS
290 FOR I=1 TO 10
300 IF F(I)>FM THEN  LET FM=F(I)
310 NEXT I
320 FOR I=1 TO 10
330 LET G(I)=F(I)*(15/FM)
340 NEXT I
350 LET B=20
360 FOR I=1 TO 10
370 LET A=21-G(I): LET A=INT (A+.5)
380 FOR Z=A TO B
390 PRINT AT Z,3*I-2;"*"
410 NEXT Z
420 NEXT I
440 FOR I=0 TO 31
450 PRINT AT 21,I;"#";
460 NEXT I
470 FOR I=1 TO 10
480 PRINT #1;(I*10);" ";
490 NEXT I
500 PRINT AT 0,8; FLASH 1; PAPER 4; INK 3;"FIN DEL PROGRAMA
": PAUSE 0
510 DATA 10.3,16.4,14.6,13.4,14.2
520 DATA 11.9,8.9,7.8,2.4,.2
```

5. PASO : Lista de variables

A = Límite inferior del grupo de edades
después : dirección de inicio de la columna

B = Límite superior del grupo de edades
después : dirección final de la columna
F = Frecuencia
FM = Frecuencia máxima
FT = Frecuencia transformada
I = Indice variable
H = Línea de pantalla

6. PASO : Descripción del programa

Línea 10-90 : Título

Línea 100-110 : Espera

Línea 120-160 : Dimensionado y lectura de los datos de partida

Línea 170-180 : Impresión del encabezamiento de la tabla

Línea 190-250 : Impresión de la tabla:

210-220 : Determinación del punto de inicio del grupo
230 : Impresión

Línea 260-270 : Espera

Línea 280-310 : Búsqueda de la frecuencia mayor y comparación de todas las frecuencias con cada frecuencia máxima alcanzada (empezando por el valor 0)

Línea 320-340 : Determinación de la frecuencia transformada mediante adaptación de las frecuencias a la frecuencia máxima

Línea 350-420 : Trazado de la gráfica:
350 : Final de la columna
370 : Determinación del comienzo de la columna medido con respecto al reparto de líneas
380-410 : Trazado de la columna mediante el encadenamiento de asteriscos

Línea 430-460 : Impresión del eje horizontal

Línea 470-490 : Escalado del eje horizontal

Línea 500 : Final del programa

Línea 510-520 : DATAS

7. PASO : Resultados

No vamos a ofrecer los resultados de este programa. Nos parece mucho más útil que el lector introduzca el programa en su ordenador y que después intente explicar paso a paso lo que sucede, comparando las impresiones en pantalla con la descripción que hacemos.

No obstante, creemos conveniente hacer la siguiente puntualización:
El gráfico que traza este programa no utiliza las instrucciones BASIC que hay disponibles para los gráficos de alta resolución, sino que se genera mediante instrucciones PRINT.

Esto tiene como consecuencia que el dibujo sea poco preciso. Las columnas de diagrama no representan exactamente las frecuencias, porque la unidad básica de trazado de las mismas es un cuadrado del tamaño del cursor, es decir con un lado de 7 mm. (en un monitor normal de ordenador).

Para obtener gráficos más exactos no queda más remedio que utilizar los gráficos de alta resolución, como haremos en el capítulo siguiente.

8.6 La economía de los paises europeos

1. PASO : Presentación del problema

El programa siguiente es muy distinto a los programas precedentes. Aquí se trata nuevamente de confeccionar un "programa informativo".

Vamos a desarrollar un programa que visualiza datos importantes sobre la economía nacional cuando el usuario lo requiere.

Estos datos se refieren a los siguientes campos:

1. Población (en millones)
2. Superficie (en miles de Km. cuadrados)
3. Producto interior bruto (en miles de millones de Ptas.)
4. Población activa (en millones)
5. Proporción de trabajadores extranjeros (en %)
6. Impuestos totales (en miles de millones de Ptas.)
7. Deudas (en miles de millones de Ptas.)
8. Porción de la producción industrial (en %)

El usuario puede alargar esta lista o utilizar otros conceptos en lugar de los países (p.e. Comunidades Autónomas).

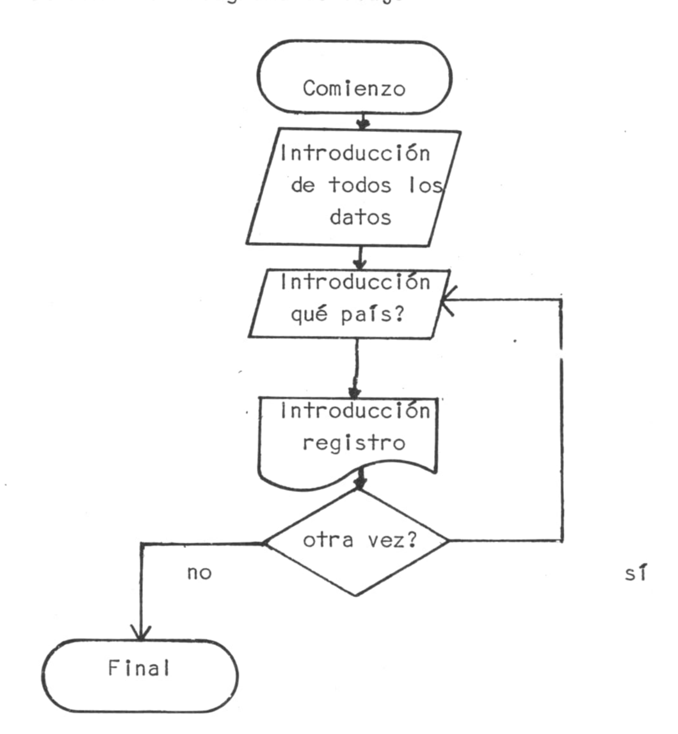
Lógicamente, habrá que modificar el fichero de datos para adaptar el programa.

2. PASO : Análisis del problema

El lector habrá podido comprobar que a lo largo de estos capítulos hay ciertos problemas que se repiten o reaparecen de forma similar a la originaria. Aquí nos encontramos de nuevo ante este problema: En principio no se presentan nuevos problemas y el análisis resulta ciertamente sencillo.

Primero hay que introducir todas las informaciones relevantes. A continuación se pregunta al usuario el nombre del país acerca del que desea obtener informaciones. Tras introducir este nombre tiene lugar la impresión del juego de datos correspondiente.

3. PASO : Diagrama de flujo



```
 10 REM E-5 ECONOMIA EUROPEA
 20 BORDER 0: PAPER 0: INK 7: CLS
 25 POKE 23658,8: POKE 23609,35
 30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA IMPRIMIR
 INFORMA-"
 40 PRINT  PAPER 4; INK 0; FLASH 1;"CIONES SOBRE  EL PODER
ECONOMICO"
 45 PRINT  PAPER 4; INK 0; FLASH 1;"DE DIFERENTES PAISES EU
ROPEOS.  "
 50 PRINT : PRINT "    ------------------------  "
 60 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
 70 PRINT "    ------------------------  "
 80 PRINT AT 16,8; FLASH 1; PAPER 2;"PULSE UNA TECLA"
 90 IF INKEY$="" THEN  GO TO 90
100 CLS
110  DIM V(8,18)
120 PRINT "DE QUE PAIS DESEA INFORMACION. ?": PRINT
130 FOR I=1 TO 18
140 READ L$
150 PRINT AT I+2,8; INK 4; BRIGHT 1;I;AT I+2,12; INK 5; BRI
GHT 1;L$
160 NEXT I
170 INPUT TAB 5; FLASH 1; PAPER 2; INK 4;"INDIQUE UN NUMERO
 ? ";Z
180 IF Z<1 OR Z>18 THEN  GO TO 170
190 CLS
200 PRINT AT 13,6;"UN MOMENTO POR FAVOR"
210 FOR J=1 TO 8
220 FOR I=1 TO 18
230 READ X
240 LET V(J,I)=X
250 NEXT I
260 NEXT J
270 RESTORE
280 FOR I=1 TO Z
290 READ L$
300 NEXT I
310 LET A$="POBLACION........(mill.)..."
320 LET B$="SUPERFICIE...(x1000 km2)..."
330 LET C$="PROD. INTERIOR...(x10E9)..."
340 LET D$="POBL. ACTIVA.....(mill.)..."
350 LET E$="PROP. EXTRANJEROS....(%)..."
360 LET F$="IMP. TOTALES.....(x10E9)..."
370 LET G$="DEUDAS...........(x10E9)..."
380 LET H$="PROD. INDUSTRIAL.....(%)..."
```

```
 390 CLS
 400 PRINT AT 0,10; FLASH 1; PAPER 2; INK 4;L$
 410 PRINT : FOR Q=0 TO 31: PRINT "-";: NEXT Q
 420 PRINT : PRINT : PRINT : PRINT A$;"  ";V(1,Z)
 430 PRINT : PRINT B$;"  ";V(2,Z)
 440 PRINT : PRINT C$;"  ";V(3,Z)
 450 PRINT : PRINT D$;"  ";V(4,Z)
 460 PRINT : PRINT E$;"  ";V(5,Z)
 470 PRINT : PRINT F$;"  ";V(6,Z)
 480 PRINT : PRINT G$;"  ";V(7,Z)
 490 PRINT : PRINT H$;"  ";V(8,Z)
 500 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A"
 510 IF INKEY$="" THEN  GO TO 510
 520 CLS
 530 PRINT AT 12,7;"OTRO PAIS MAS (S/N)"
 540 LET O$=INKEY$
 550 IF O$="S" THEN  CLS : RESTORE : GO TO 120
 560 IF O$="N" THEN  CLS : GO TO 580
 570 GO TO 540
 580 PRINT AT 11,7; PAPER 2; INK 4; FLASH 1;"FIN DEL PROGRAM
A": PAUSE 0
 600 DATA "ESPA/A","FRANCIA","ITALIA"
 610 DATA "REINO UNIDO","RFA","IRLANDA"
 620 DATA "GRECIA","BELGICA","HOLANDA"
 630 DATA "SUECIA","DINAMARCA","PORTUGAL"
 640 DATA "NORUEGA","FINLANDIA","AUSTRIA"
 650 DATA "LUXEMBURGO","AUSTRIA","MALTA"
 660 DATA 0,0,0,0,0,0,0,0,0,0
 670 DATA 0,0,0,0,0,0,0,0,0,0
 680 DATA 0,0,0,0,0,0,0,0,0,0
 690 DATA 0,0,0,0,0,0,0,0,0,0
 700 DATA 0,0,0,0,0,0,0,0,0,0
 710 DATA 0,0,0,0,0,0,0,0,0,0
 720 DATA 0,0,0,0,0,0,0,0,0,0
 730 DATA 0,0,0,0,0,0,0,0,0,0
 740 DATA 0,0,0,0,0,0,0,0,0,0
 750 DATA 0,0,0,0,0,0,0,0,0,0
 760 DATA 0,0,0,0,0,0,0,0,0,0
 770 DATA 0,0,0,0,0,0,0,0,0,0
 780 DATA 0,0,0,0,0,0,0,0,0,0
 790 DATA 0,0,0,0,0,0,0,0,0,0
 800 DATA 0,0,0,0,0,0,0,0,0,0
 810 DATA 0,0,0,0,0,0,0,0,0,0
 820 DATA 0,0,0,0,0,0,0,0.0.0,0
 830 DATA 0,0,0,0,0,0,0,0,0,0
```

5. PASO : Lista de variables

A$ = Variable de cadena
B$ = Denominaciones de variables
I = Variable índice
J = Variable índice
L$ = Nombre de los países
V = Variables
X = Campo de datos
Z = Número del país buscado

6. PASO : Descripción del programa

Línea 10-90 : Título y espera

Línea 100-110 : Borrado y dimensionado

Línea 120-180 : Requirimiento del número del país del cual se necesitan informaciones después de ofrecer una lista de países en la pantalla.

Línea 190-260 : Borrado de la pantalla y lectura de los datos

Línea 270 : Inicializa el puntero de datos

Línea 280-300 : Búsqueda del país seleccionado

Línea 310-390 : Introducción de los nombres de las variables y borrado de la pantalla

Línea 400-410 : Impresión del nombre del país

Línea 420-490 : Impresión del registro

Línea 500-520 : Espera y borrado de la pantalla

Línea 530-580 : Consulta si desea otra información (en caso afirmativo, borrar la pantalla y volver a la línea 120, en caso negativo final del programa)

Línea 600-650 : DATAS de los nombres de los países

Línea 660-830 : Valores de las variables: introducir los valores de tal manera, que para la primera variable se den todos los valores sobre el primer país, para la segunda variable todos los valores sobre el segundo país,... etc.

ATENCION : EN EL PROGRAMA APARECEN SOLO CEROS (¡ver siguiente paso!)

7. PASO : Resultados

El fichero de datos de este programa contiene únicamente ceros, de manera que si hacemos una pasada de prueba, sólo aparecerán ceros en la pantalla.

Para efectuar una pasada ''auténtica'', hay que introducir, primero, datos concretos, que el lector puede tomar de la tabla siguiente.

Datos económicos de diferentes países europeos :

País	Pobl. (mill.)	Superf. (1000 km2)	Produc. Interior (10E9)	Poblac. activa (mill.)
España				
Francia				
Italia				
Reino U.				
RFA				
Irlanda				
Grecia				
Bélgica				
Holanda				
Suecia				
Dinamarca				
Portugal				
Noruega				
Finlandia				
Suiza				
Luxemburgo				
Austria				
Malta				

País	Prop. extranjeros	Impues. totales (10E9)	Deudas. (10E9) Ptas.	Produc. indust. (%)
España				
Francia				
Italia				
Reino U.				
RFA				
Irlanda				
Grecia				
Bélgica				
Holanda				
Suecia				
Dinamarca				
Portugal				
Noruega				
Finlandia				
Suiza				
Luxemburgo				
Austria				
Malta				

Capítulo 9 : Matemáticas II



9.1 Consideración previa

Para finalizar incluimos un capítulo en el que se tratan nuevos problemas matemáticos, con un grado de dificultad algo mayor que los vistos hasta ahora. De esta forma, el usuario experimentado en la utilización del microordenador, comprobará que también se pueden abordar problemas de mayor complejidad con programas relativamente sencillos.
Llegados a este punto, podría parecer que el capítulo final se halla un poco por encima de las posibilidades del principiante (pero es precisamente esto lo que a veces le proporciona más interés).

Estos problemas de mayor complejidad, deben ser programados de tal forma que no requieran instrucciones BASIC adicionales. Esto significa, que los programas resultantes (al igual que todos los programas precedentes) no pueden satisfacer los niveles de elegancia de los programadores experimentados. Pero esto aquí no tiene importancia.

En contraposición a los anteriores capítulos, aquí seremos más breves, pues se van a tratar diecisiete nuevos problemas:

Después de una pequeña presentación del problema y de algunas indicaciones con referencia a su análisis, pasamos inmediatamente a presentar el programa y terminamos con una pequeña descripción del mismo. En adelante prescindiremos de los pasos "Diagrama de flujo" y "Resultados", no sólo por motivos de espacio, sino porque en cuanto al aspecto de la programación pura no introducen nuevos problemas de detalle.

9.2 Red

Presentación del problema

El programa siguiente sirve para trazar en la pantalla una red de líneas con separaciones entre líneas y columnas variable.
El esquema mental utilizado como base de estos programas, sirve de forma parecida por ejemplo para estructurar tablas extensas con líneas horizontales y verticales.

Análisis del problema

Podemos ahorrarnos cualquier explicación acerca del análisis del problema. El usuario únicamente debe introducir las distancias entre líneas y entre columnas. El trazado de las líneas se efectúa mediante dos bucles FOR-NEXT y la instrucción BASIC ''DRAW''.

```
  10 REM M-II    -RETICULA-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA TRAZAR U
NA RED EN"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"EL MONITOR DE SU ORDENA
DOR.      "
  50 PRINT : PRINT "    -----------------------   "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "   -----------------------  "
  70 PRINT AT 13,0;"DISTANCIA ENTRE COLUMNAS (C<256)";: INPU
T TAB 12;"C=? ";S
  80 PRINT AT 13,0;"DISTANCIA ENTRE FILAS    (F<175)";: INPU
T TAB 12;"F=? ";Z
  90 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A"
 100 IF INKEY$="" THEN  GO TO 100
 110 CLS
 120 FOR I=0 TO 175 STEP Z
 130 PLOT 0,I: DRAW 255,0
 140 NEXT I
 150 FOR I=0 TO 255 STEP S
 160 PLOT I,0: DRAW 0,175
 170 NEXT I
 180 PRINT #0;TAB 8; INK 0; FLASH 1; PAPER 5;"FIN DEL PROGRA
MA": PAUSE 0
```

Lista de variables:

I = Variable índice

S = Separación entre columnas

Z = Separación entre líneas

Descripción del programa:

Línea 10-60 : Título

Línea 70-80 : Introducción de la distancia entre líneas y entre columnas

Línea 90-110 : Espera una entrada desde el teclado y borra la pantalla

Línea 120-140 : Trazado de las líneas horizontales

Línea 150-170 : Trazado de las líneas verticales

Línea 180 : Final del programa

9.3 La recta

Presentación del problema

El programa siguiente sirve para dibujar cualquier recta en un eje de coordenadas cuyo origen se sitúa en el centro de la pantalla (127,87).

Análisis del problema

La posición de una recta queda determinada por dos parámetros, la ordenada en el origen A y la tangente del ángulo que forma con el eje de ordenadas B, según la siguiente ecuación:

$$Y_i = A + B*X_i$$

Dado que el origen de coordenadas debe estar situado en (127,87), resulta que:

$$Y_i = 87 + A + B*(X_i - 127)$$

En todo caso, debemos tener en cuenta que en contraposición al eje de coordenadas normal, los valores de Y empiezan en la parte superior de la pantalla (la línea número 0 es la línea superior de la pantalla).

Por esta razón, debe modificarse la función de la forma siguiente:

```
  10 REM M-II/2    -RECTA-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA  TRAZAR
UNA LINEA"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"RECTA.
      "
  50 PRINT : PRINT "    ------------------------  "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------  "
  70 PRINT AT 11,4;"ABCISA EN EL ORIGEN  : ";: INPUT TAB 12;
"A=? ";A: PRINT A
  80 PRINT AT 16,4;"PENDIENTE DE LA RECTA: ";: INPUT TAB 12;
"P=? ";B: PRINT B
  90 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A"
 100 IF INKEY$="" THEN  GO TO 100
 110 CLS
 120 PLOT 0,87: DRAW 255,0
 130 PLOT 127,0: DRAW 0,175
 140 FOR X=0 TO 255
 150 LET Y=87-A-B*(X-127)
 160 IF Y<=0 OR Y>=175 THEN  GO TO 180
 170 PLOT  INK 5; BRIGHT 1;X,Y
 180 NEXT X
 190 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"FIN DEL PROGRA
MA": PAUSE 0
```

Lista de variables:

A = Abcisa en el origen

B = Tangente del ángulo de pendiente

I = Contador

X = Valores sobre el eje X

Y = Valores de la función

Descripción del programa:

Línea 10-60 : Título

Línea 70-80 : Introducción de los parámetros de la recta

Línea 90-110 : Espera y borrado de la pantalla

Línea 120-130 : Trazado de los ejes de coordenadas

Línea 140-180 : Trazado de la recta sobre todo el ancho de la pantalla (si los valores de Y son no válidos, se salta la instrucción)

Línea 190 : Final del programa

9.4 El círculo

Presentación del problema

Siguiendo el mismo patrón del punto anterior, donde se desarrolló un programa para trazar una recta, de nuevo presentamos un programa para dibujar todo tipo de círculos. Aquí aparecen problemas ya tratados, de modo que seremos breves.

Análisis del problema

Nos enfrentamos aquí al mismo problema que en el punto anterior, con la diferencia de que aquí tomamos como base la ecuación del círculo:

Esta se expresa de la forma siguiente:

$$(x - a)^2 + (y - b)^2 = r^2$$

r es el radio del círculo, x e y son las coordenadas de la línea circular y a y b son las coordenadas del centro del círculo.

Si despejamos y (resolución de una ecuación cuadrática), obtenemos las ecuaciones para el semicírculo superior e inferior:

$$(y - b)^2 = r^2 - (x - a)^2 = D$$

$$- b = \pm \text{ SQR (D)}$$

$$y = \pm \text{ b SQR (D)}$$

Con ello hemos aclarado los aspectos fundamentales de este programa:

Requiere como informaciones de entrada las coordenadas del centro del círculo a y b (que a partir de ahora llamaremos T y Z) y el radio r (en BASIC : R). A partir de aquí podrán determinarse los valores Y1 e Y2 correspondientes a diferentes valores de X y después trazar los puntos X,Y1 y X,Y2.

Debe prestarse atención a que X se mueva únicamente en el ámbito que va de T-R a T+R, porque fuera de esta zona no hay valores de Y definidos dentro del campo real (el valor auxiliar D sería negativo, y al tratar de extraer la raiz del mismo provocaría un mensaje de error y una interrupción del programa.

```
  10 REM M-II/3  -CIRCULO-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA TRAZAR UN
A CIRCUN-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"FERENCIA.
       "
  50 PRINT : PRINT "    -----------------------  "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    -----------------------  "
  70 PRINT AT 10,0;"COORDENADA X DEL ORIGEN     : ";: INPUT T
AB 12;"X= ? ";T: PRINT T
  80 PRINT AT 14,0;"COORDENADA Y DEL ORIGEN     : ";: INPUT T
AB 12;"Y= ? ";Z: PRINT Z
  90 PRINT AT 18,0;"RADIO DE LA CIRCUNFERENCIA : ";: INPUT T
AB 12;"R= ? ";R: PRINT R
 100 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A"
 110 IF INKEY$="" THEN  GO TO 110
 120 CLS
 130 PLOT 0,87: DRAW 255,0
 140 PLOT 127,0: DRAW 0,175
 150 PLOT 0,87+Z: DRAW  INK 5;255,0
 160 PLOT 127+T,0: DRAW  INK 5;0,175
 170 LET A=T-R: LET B=T+R
 180 FOR X=A TO B
 190 LET D=(R*R)-((X-T)*(X-T))
 200 LET Y=Z+SQR D
 210 LET M=X+127: LET N=Y+87
 220 IF M>=255 THEN  LET M=255
 230 IF N>=175 THEN  LET N=175
 240 IF M<=0 THEN  LET M=0
 250 IF N<=0 THEN  LET N=0
 260 PLOT  INK 6;M,N
 270 NEXT X
 280 FOR X=B TO A STEP -1
 290 LET D=(R*R)-((X-T)*(X-T))
 300 LET Y=Z-SQR D
 310 LET M=X+127: LET N=Y+87
 320 IF M>=255 THEN  LET M=255
 330 IF N>=175 THEN  LET N=175
 340 IF M<=0 THEN  LET M=0
 350 IF N<=0 THEN  LET N=0
 360 PLOT  INK 6;M,N
 370 NEXT X
 380 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"FIN DEL PROGRA
MA": PAUSE 0
```

Lista de variables

A = Punto extremo izquierdo del círculo
B = Punto extremo derecho del círculo
D = D = R - (X-T)
I = Indice variable
R = Radio
T = Columna del centro del círculo
X = Valores de coordenada X
Y = Valores de coordenada Y
Z = Línea del centro del círculo

Descripción del programa

Línea 10-60 : Título y aclaraciones

Línea 70-120 : Informaciones y borrado de la pantalla

Línea 130-140 : Eje de coordenadas

Línea 150-160 : Eje de coordenadas auxiliar

Línea 170 : Fijación del ámbito de valores de X

Línea 180-270 : Trazado de un semicírculo

Línea 280-370 : Trazado del otro semicírculo

Línea 380 : Final del programa

9.5 La sinusoide

Presentación del problema

En este último ejemplo de gráfico de alta resolución vamos a representar una oscilación sinusoidal, como la que ya nos encontramos en el capítulo ''Física''. El procedimiento a seguir es el mismo que en los dos puntos precedentes.

Análisis del problema

El análisis del problema es muy sencillo, y no se presentan cuestiones nuevas. Unicamente debe procurarse adaptar la periodicidad de la oscilación (frecuencia) y su amplitud (valor máximo) de forma que la representación gráfica en la pantalla sea ópticamente correcta, es decir, que el espacio disponible sea aprovechado de forma hábil.

```
  10 REM M-II/4  -SENO-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA TRAZAR UN
A SINUSOI"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"DE.
      "
  50 PRINT : PRINT "     ------------------------  "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------  "
  70 PRINT AT 11,0;"FRECUENCIA (funcion de pi): ";: INPUT TA
B 8;"Frecuencia = ? ";P: PRINT P
  80 PRINT AT 17,0;"AMPLITUD  (de 0 a 87)      : ";: INPUT TA
B 8;"Amplitud = ? ";A: PRINT A
  90 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A"
 100 IF INKEY$="" THEN  GO TO 110
 110 CLS
 130 PLOT 0,87: DRAW 255,0
 140 FOR X=0 TO 255
 150 LET J=X/50: LET J=J*(1/P)
 160 LET Y=SIN J: LET Y=Y*A
 170 LET Y=87-Y
 180 PLOT  INK 5; BRIGHT 1;X,Y
 190 NEXT X
 200 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"FIN DEL PROGRA
MA": PAUSE 0
```

Lista de variables:

A = Amplitud
I = Indice variable
J = Valor X transformado
P = Frecuencia
X = Valor de la abcisa X
Y = Valor de la ordenada Y

Descripción del programa

Línea 10-60 : Título

Línea 70-110 : Informaciones INPUT y borrado pantalla

Línea 130 : Trazado de una línea central horizontal

Línea 140-190 : Determinación de los valores de la función y trazado después de transformar de acuerdo con P y A

Línea 200 : Impresión texto "FIN DEL PROGRAMA"

9.6 Tabla matemática 1

Presentación del problema

En el programa siguiente se imprimen los cuadrados y las raíces cuadradas de todos los números naturales del 1 al 100.
Este programa debe ser entendido casi como "programa de consulta".

Análisis del problema

Siendo tan sencillo el problema, pensamos que resulta superfluo cualquier análisis.

```
  10 REM M-II/5  -TABLA 1-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA GENERAR L
OS CUADRA"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"DOS Y LAS RAICES CUADRA
DAS DE TO"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"DOS LOS NUMEROS NATURAL
ES(1-100)"
  50 PRINT : PRINT "    -----------------------  "
  55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    -----------------------  "
  64 LET CONTROL=4
  65 GO SUB 2000: REM ESPERA
  70 FOR I=1 TO 100
  80 LET Q=I*I: LET W=SQR I: LET W=INT (W*1000+.5)/1000
  90 PRINT AT CONTROL,4;I;AT CONTROL,14;Q;AT CONTROL,26;W
  95 LET CONTROL=CONTROL+1
  96 IF CONTROL>20 THEN  LET CONTROL=4
 100 IF I/17=INT (I/17) THEN  GO SUB 2000
 110 NEXT I
 120 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"FIN DEL PROGRA
MA": PAUSE 0: STOP
1000 REM UP CABECERA
1100 PRINT AT 0,2;"NUMERO";AT 0,12;"CUADRADO";AT 0,26;"RAIZ"
1200 PRINT : FOR A=0 TO 31: PRINT "-";: NEXT A
1300 RETURN
2000 REM UP ESPERA
2010 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A"
2020 IF INKEY$="" THEN  GO TO 2020
2030 CLS : GO SUB 1000
2040 RETURN
```

Lista de variables:

I = Indice variable (números naturales 1-100)
J = Indice variable
Q = Número cuadrado
W = Raiz cuadrada

Descripción del programa

Línea 10-60 : Título

Línea 65 : Salto a la subrutina 2000 para la espera

Línea 70-110 : Cálculo e impresión de los valores en cuestión

Línea 100: Después de cada 17 líneas salto a la subrutina 2000 para la espera

Línea 120 : Final del programa principal

Línea 1000-1300 : Subrutina para la impresión de la cabecera de la tabla

---

Línea 2000-2040 : Subrutina para la espera

Línea 2010 : Impresión de un mensaje
Línea 2020 : Espera
Línea 2030 : Borrado de la pantalla, impresión de una nueva cabecera mediante el salto a la subrutina 1000
Línea 2040 : Retorno al programa principal

9.7 Tabla matemática 2

Presentación del problema

Este programa sirve para ofrecer una tabla de valores de las funciones

- seno (SIN)
- coseno (COS)

Estas funciones angulares se representan en dependencia del parámetro (llamado pi en el programa). Por esto, en la tabla resultante aparecen también valores en grados angulares.

Análisis del problema

Tampoco aquí resulta necesario el análisis del problema, dado que no se presentan especiales dificultades.

```
 10 REM M-II/6  -TABLA 2-
 20 BORDER 0: PAPER 0: INK 7: CLS
 25 POKE 23658,8: POKE 23609,35
 30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA DETERMINA
R SENOS Y"
 40 PRINT  PAPER 4; INK 0; FLASH 1;"COSENOS.
 "
 50 PRINT : PRINT "    ------------------------   "
 55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
 60 PRINT "    ------------------------   "
 65 PRINT AT 16,8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECLA
"
 70 IF INKEY$="" THEN  GO TO 70
 80 GO SUB 220
 90 FOR J=0 TO 65
100 LET I=J/10: LET I=INT (I*100+.5)/100
110 LET A=(360*I)/(2*PI): LET A=INT (A+.5)
120 LET B=SIN I: LET B=INT (B*1000+.5)/1000
130 LET C=COS I: LET C=INT (C*1000+.5)/1000
140 PRINT AT Z,2;I;AT Z,10;A;AT Z,16;B;AT Z,25;C
150 LET Z=Z+2
160 IF Z>20 THEN  LET Z=5
170 IF J=0 THEN  GO TO 190
180 IF J/7=INT (J/7) THEN  PRINT #0;TAB 8; PAPER 2; INK 4;"
PULSE UNA TECLA": PAUSE 0: GO SUB 220
190 NEXT J
200 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A": PAUSE 0: CLS
210 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"FIN DEL PROGRA
MA": PAUSE 0: STOP
220 CLS : LET Z=5: PRINT  INK 4; BRIGHT 1;"  ";"PI";"    ";
"GRADOS";"   ";"SIN";"      ";"COS"
230 PRINT
240 FOR X=0 TO 31
250 PRINT  INK 3; BRIGHT 1;"-";
260 NEXT X
270 RETURN
```

Lista de variables:

A = Angulo (en grados)
B = Seno
C = Coseno
I = J/10 (=abcisa de la función angular)
J = Indice variable (aquí 0-40)

Descripción del programa:

Línea 10-60 : Título

Línea 65-80 : Bucle de espera y salto a la subrutina de impresión de la cabezera.

Línea 90-190 : Cálculos e impresión de los resultados

Se determinan 66 valores (J=0 TO 65) siendo aplicadas las funciones angulares sobre I = J/10 (línea 100)

Línea 110 : Transformación en grados
Línea 120-130 : Determinación del SIN y del COS

Línea 140 : Impresión

Línea 150-190 : Bucle contador del número de líneas que se imprimen.(Si se imprimen más de 8 líneas se borra la panlla, se salta a la subrutina se impresión de la tabla y se siguen los cálculos).

Línea 200-210 : Final del programa principal

Línea 220-270 : Subrutina para la impresión de la cabecera de la tabla

---

9.8 Tabla matemática 3

Presentación del problema

El programa siguiente sirve para imprimir los logaritmos neperianos ln (X) dentro del ámbito de valores de X que va de 1 a 100.

Análisis del problema

Debido a la sencillez del problema también prescindimos aquí de cualquier análisis.

```
  10 REM M-II/7  -TABLA 3-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA DETERMINA
R LOS VA-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"LORES DE LOS LOGARITMOS
 NEPERIA-"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"NOS.
        "
  50 PRINT : PRINT "    -----------------------   "
  55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    -----------------------   "
  70 PRINT AT 16,8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECLA
"
  80 PAUSE 0
  90 GO SUB 200
 100 FOR I=1 TO 100
 110 LET L=LN I
 120 LET L=INT (L*10000+.5)/10000
 130 PRINT AT Z,7;I;AT Z,19;L
 140 LET Z=Z+2
 150 IF Z>20 THEN  LET Z=4
 160 IF I/8=INT (I/8) THEN  PRINT #0;TAB 8; FLASH 1; PAPER 2
; INK 4;"PULSE UNA TECLA": PAUSE 0: GO SUB 200
 170 NEXT I
 180 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A": PAUSE 0: CLS
 190 INPUT 0: PRINT AT 11,8; FLASH 1; PAPER 2; INK 4;"FIN DE
L PROGRAMA": PAUSE 0: STOP
 200 CLS : LET Z=4: PRINT  INK 4; BRIGHT 1;"      ";"NUMERO";
"       ";"LOGARITMO"
 210 PRINT
 220 FOR X=0 TO 31
 230 PRINT  INK 3; BRIGHT 1;"-";
 240 NEXT X
 250 RETURN
```

Lista de variables :

I = Indice variable (aquí 1-100)

L = Logaritmo neperiano de I

Descripción del programa:

Línea 10-60 : Título

Línea 70-80 : Bucle de espera.

Línea 90 : Salto a la subrutina de impresión de la tabla.

Línea 100-170 : Cálculo de los logaritmos e impresión, después de cada 8 líneas de la tabla, esperar

Línea 180-190 : Final del programa principal

Línea 200-250 : Subrutina para la impresión de la cabecera de la tabla

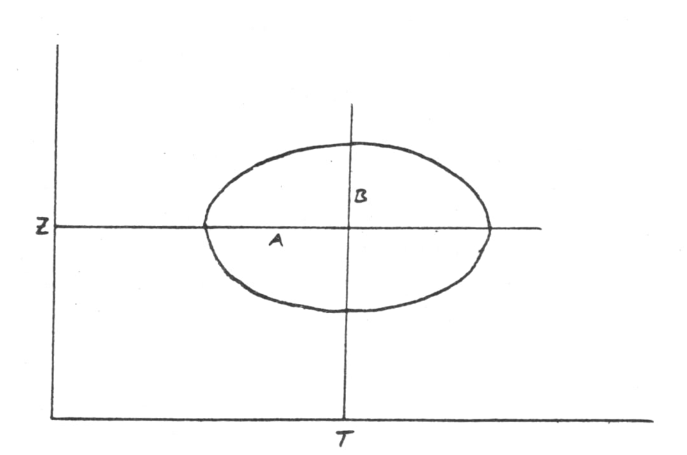
---

9.9 Elipse

Presentación del problema

El programa dibuja una elipse. Para ello deben introducirse los siguientes valores.

T: Columna del centro de la elipse (0-319)
Z: Línea del centro de la elipse (0-159)
A: Primer semieje de la elipse
B: Segundo semieje de la elipse



Análisis del problema

La ecuación de la elipse en la llamada forma central (centro de la elipse en el centro del eje de coordenadas):

$$\frac{x^2}{a} + \frac{y^2}{b} = 1$$

Para poder dibujar una función como esta mediante un programa BASIC, debemos despejar y, resultando:

$$y_{1/2} = \pm \frac{b}{a} \sqrt{a^2 - x^2}$$

Si tenemos en cuenta que una elipse no sólo puede tener su punto central en la coordenada (0,0), sino también en una cualquiera (T,Z), resultan necesarias correcciones mediante traslación del mismo (ver la siguiente descripción de programa).

```
  10 REM M-II/8  -ELIPSE-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA EL TRAZAD
O DE ELIP"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"SES.
      "
  50 PRINT : PRINT "    ------------------------   "
  55 PRINT AT 4,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------   "
  65 PRINT AT 8,0; PAPER 1; INK 3; BRIGHT 1;"* COORDENADAS D
EL CENTRO :"
  70 PRINT AT 10,2;"-Coordenada x (columna)..: ";: INPUT TAB
 12;"x= ? ";T: PRINT T
  80 PRINT AT 12,2;"-Coordenada y (fila).....: ";: INPUT TAB
 12;"y= ? ";Z: PRINT Z
  90 PRINT AT 15,0; PAPER 1; INK 3; BRIGHT 1;"* LONGITUD DE
LOS EJES  : "
 100 PRINT AT 17,2;"- 1. Semieje............: ";: INPUT TAB
 12;"? = ";A: PRINT A
 110 PRINT AT 19,2;"- 2. Semieje............: ";: INPUT TAB
 12;"? = ";B: PRINT B
 120 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A": PAUSE 0: CLS
 130 PLOT 0,87: DRAW 255,0
 140 PLOT 127,0: DRAW 0,175
 150 PLOT 0,87+Z: DRAW  INK 5;255,0
 160 PLOT 127+T,0: DRAW  INK 5;0,175
 170 LET C=T-A: LET D=T+A
 180 FOR X=C+1 TO D-1
 190 LET DS=(B/A)*SQR ((A*A)-((X-T)*(X-T)))
 200 LET Y=Z+DS
 210 LET M=X+127: LET N=Y+87
 220 IF M>=255 THEN  LET M=255
 230 IF N>=175 THEN  LET N=175
 240 IF M<=0 THEN  LET M=0
 250 IF N<=0 THEN  LET N=0
 260 PLOT  INK 6;M,N
 270 NEXT X
 280 FOR X=D-1 TO C+1 STEP -1
 290 LET DS=(B/A)*SQR ((A*A)-((X-T)*(X-T)))
 300 LET Y=Z-DS
 310 LET M=X+127: LET N=Y+87
 320 IF M>=255 THEN  LET M=255
 330 IF N>=175 THEN  LET N=175
 340 IF M<=0 THEN  LET M=0
 350 IF N<=0 THEN  LET N=0
 360 PLOT  INK 6;M,N
 370 NEXT X
 380 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"FIN DEL PROGRA
MA": PAUSE 0
```

Lista de variables:

A = Primer semieje de la elipse
B = Segundo semieje de la elipse
C = Extremo izquierdo de la elipse
D = Extremo derecho de la elipse
DS = Valor auxiliar tomado de la ecuación de la elipse
I = Indice variable
T = Columna del punto central de la elipse
X = Abcisa
Y = Ordenada
Z = Línea del punto central de la elipse

Descripción del programa

Línea 10-60 : Título

Línea 65-110 : Fijación de las coordenadas del punto central y de los semiejes

Línea 120 : Espera y borrado de pantalla

Línea 130-140 : Trazado de los ejes de coordenadas

Línea 150-160 : Trazado de los ejes auxiliares

Línea 170-270 : Trazado de la mitad inferior de la elipse

Línea 280-370 : Trazado de la mitad superior de la elipse

Línea 380 : Final del programa

9.10 Distribución normal de Gauss

Presentación del problema

Este programa representa gráficamente la distribución de Gauss.
Tiene la función de densidad

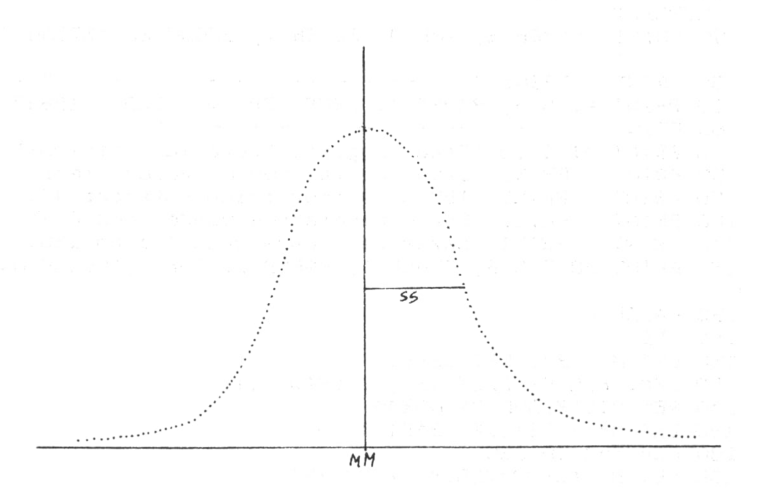
$$f(x) = \frac{1}{SS\sqrt{2\pi}} \cdot e^{\left[-\frac{1}{2}\left(\frac{X - MM}{SS}\right)^2\right]}$$

MM y SS son los parámetros de posición caracterizadores (ver el esquema siguiente):

MM designa el valor medio de la distribución
SS designa la dispersión de la distribución
(desviación estandar = distancia entre la línea central y cualquiera de los dos puntos de inflexión de la distribución).

En este programa seleccionado los siguientes valores :

MM = 127
SS = 40



Análisis del problema

Nuevamente prescindible.

```
  10 REM M-II/9  -GAUSS-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"ALGORITMO PARA REPRESEN
TAR GRAFI"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"CAMENTE LA  DISTRIBUCIO
N  NORMAL"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"SEGUN EL METODO DE GAUS
S.       "
  50 PRINT : PRINT "     -----------------------   "
  55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    -----------------------   "
  70 PRINT AT 10,0;"Este programa traza la distribu-"
  80 PRINT : PRINT "cion  normal con el valor  medio"
  90 PRINT : PRINT "127 y la desviacion estandar 40."
 100 PRINT : PRINT "Estos parametros pueden modifi-"
 110 PRINT : PRINT "carse para cada caso (linea 150)"
 120 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A"
 130 PAUSE 0
 140 CLS
 150 LET MM=127: LET SS=40
 160 DRAW 255,0: PLOT 127,0: DRAW 0,175
 170 REM DISTRIBUCION NORMAL
 180 LET A=1/(SS*SQR (2*PI))
 190 FOR X=0 TO 255
 200 LET B=((X-MM)/SS)*((X-MM)/SS)
 210 LET B=.5*B*(-1)
 220 LET Y=A*EXP B
 230 LET Y=(600000/SS)*Y
 240 LET Y=175-Y
 250 IF Y<0 OR Y>175 THEN  GO TO 270
 260 PLOT  INK 3; BRIGHT 1;X,Y-175
 270 NEXT X
 280 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"FIN DEL PROGRA
MA": PAUSE 0
```

Lista de variables:

A = Primer término de la ecuación funcional
B = Exponente en la ecuación funcional
I = Indice variable
MN = Valor medio de la distribución normal
SS = Desviación estandar de la distribución normal
X = Abcisa
Y = Ordenada

Descripción del programa

Línea 10-120 : Título y aclaraciones

Línea 130-140 : Espera y borrado de la pantalla

Línea 150 : Definición de los parámetros de la función de distribución y modo gráfico

Línea 160 : Trazado de unos ejes de coordenadas

Línea 170-270 : Cálculo de los valores de ordenada de la distribución normal, normalización (línea 230) y trazado de la curva

Línea 290-310 : Final del programa

9.11 Regresión

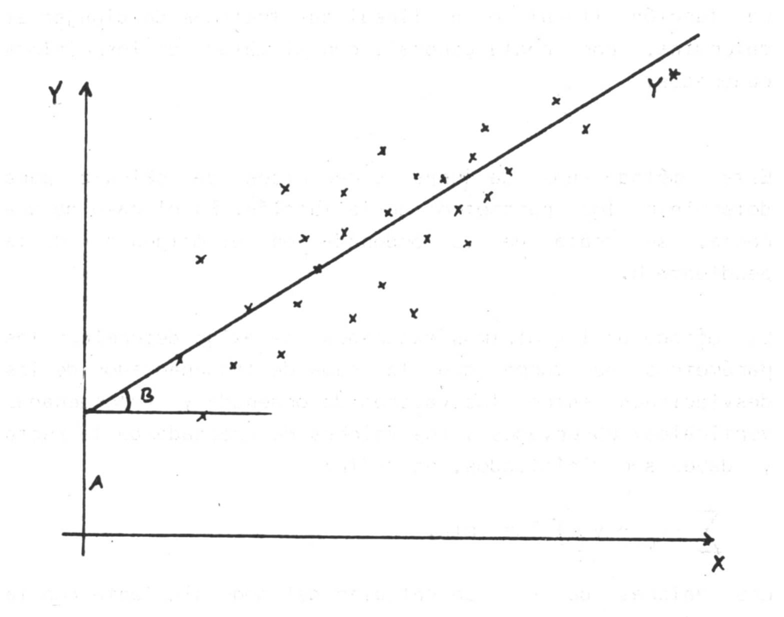
Presentación del problema

El cálculo de regresiones constituye una cuestión importante dentro del campo de la estadística matemática. Se trata de describir una distribución estadística bivariable mediante una función matemática.

Una distribución estadística bivariable se produce, por ejemplo, cuando estudiamos simultaneamente dos características de una población, es decir, si consultamos a una serie de personas acerca de su tamaño y de su peso al mismo tiempo.

El objetivo de una investigación como ésta, es descubrir si existe alguna relación entre ambas variables.

Una distribución bivariable puede representarse gráficamente en un sistema de coordenadas bidimensional por medio de una "nube de puntos", como puede apreciarse en la página siguiente.



Para describir el tipo de relación existente entre ambas variables, o para determinar la tendencia lineal en una serie de tiempos, que desde el punto de vista matemático es el mismo problema, se intenta encajar de la mejor forma posible una función matemática lineal entre los puntos.

Del mismo modo, existen directrices de cálculo no lineales, pero que aquí no serán tratadas.

Análisis del problema

La función lineal o no lineal que tratamos de dibujar se determina, por regla general, con el método de los mínimos cuadrados.

Este método nos da unas directrices de cálculo para determinar los parámetros de la función. En el caso de una recta, se trata de la ordenada en el origen A y de la pendiente B.

El método de los mínimos cuadrados nos exige determinar los parámetros de forma que la suma de los cuadrados de las desviaciones entre los valores de ordenada y (coordenadas verticales) observados y los valores de ordenada de la recta y deben ser minimizados, es decir:

$$\sum (y_i - y_i)^2 = \text{min.}$$

Los valores de y se calculan del modo siguiente con la ecuación de la recta:

$$y_i = a + b * x_i$$

Con la regla de minimización antes citada resulta:

$$\sum (y_i - a - b * x_i)^2 = \min.$$

El mínimo de esta función se halla realizando una derivada parcial según a y b e igualando estas dos derivadas parciales a cero.

De esta forma se obtienen dos ecuaciones con dos incógnitas (a y b), en las que éstas se habrán de despejar. Se obtienen así las dos ecuaciones:

$$b = \frac{n * \sum x_i y_i - \sum x_i \sum y_i}{n * \sum x_i^2 - (\sum x_i)^2}$$

$$a = \frac{\sum y_i}{n} - b * \frac{\sum x_i}{n}$$

El programa BASIC a desarrollar sirve para calcular estos parámetros según el método de los mínimos cuadrados para cualquier par de valores x ,y que se introduzca y para a continuación imprimir los resultados.

```
  10 REM M-II/10  -REGRESION-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA DETERMINA
R LOS PA-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"RAMETROS  DE UNA FUNCIO
N  LINEAL"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"DE REGRESION.
       "
  50 PRINT : PRINT "     ------------------------   "
  55 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  60 PRINT "    ------------------------   "
  70 PRINT AT 10,0;" Este programa requiere pares de"
  80 PRINT : PRINT "valores como informaciones de en"
  90 PRINT : PRINT "trada (P.E. valores x e y, o va-"
 100 PRINT : PRINT "lores de tiempo y caracteristica"
 110 PRINT : PRINT "etc......)"
 120 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 4;"PULSE UNA TECL
A"
 130 PAUSE 0
 140 CLS
 150 PRINT AT 12,2;"CUANTOS PARES DE VALORES : ";: INPUT TAB
 12;"? = ";N: PRINT N
 160 DIM X(N): DIM Y(N)
 170 CLS
 180 PRINT AT 2,0; FLASH 1; INK 4; PAPER 2; BRIGHT 1;"-INTRO
DUZCA LOS PARES DE VALORES"
 190 FOR I=1 TO N
 200 PRINT AT 6,10; PAPER 1; INK 6; BRIGHT 1;"PAR NUMERO   "
;I
 205 PRINT : FOR M=0 TO 31: PRINT "-";: NEXT M
 210 PRINT AT 12,8;"VALOR DE X......: ";: INPUT TAB 13;"X =
";Z1: PRINT Z1
 220 PRINT AT 16,8;"VALOR DE Y......: ";: INPUT TAB 13;"Y =
";Z2: PRINT Z2
 230 LET X(I)=Z1: LET Y(I)=Z2
 235 CLS
 240 NEXT I
 250 GO SUB 560
 260 FOR I=1 TO N
 270 PRINT AT P,3;I;AT P,13;X(I);AT P,24;Y(I)
 280 IF P=16 THEN  PRINT #0;TAB 8; PAPER 3; INK 4; FLASH 1;
BRIGHT 1;"PULSE UNA TECLA": PAUSE 0: GO SUB 560
 290 LET P=P+2
 300 NEXT I
```

```
 310 PRINT #0;TAB 9; PAPER 2; INK 5; BRIGHT 1;"FIN DEL CONTR
OL"
 320 PAUSE 0: CLS
 325 LET S1=0: LET S2=0: LET S3=0: LET S4=0
 330 FOR I=1 TO N
 340 LET S1=S1+X(I)*Y(I)
 350 LET S2=S2+X(I): LET S3=S3+Y(I)
 360 LET S4=S4+X(I)*X(I)
 370 NEXT I
 380 LET B=(N*S1-S2*S3)/(N*S4-S2*S2)
 390 LET A=S3/N-B*S2/N
 400 LET B=INT (B*1000+.5)/1000
 410 LET A=INT (A*1000+.5)/1000
 420 PRINT AT 4,12; INK 2; INK 5; BRIGHT 1;"RESULTADO"
 430 PRINT AT 10,0;"ORDENADA              A = ";A
 440 PRINT AT 14,0;"PENDIENTE             B = ";B
 450 PRINT #0;TAB 8; PAPER 3; INK 4; FLASH 1; BRIGHT 1;"PULS
E UNA TECLA"
 460 PAUSE 0
 470 CLS
 480 PRINT AT 4,0;" Con estos parametros puede lla-"
 490 PRINT : PRINT "marse al programa para el traza-"
 500 PRINT : PRINT "do de una  recta, si desea obte-"
 510 PRINT : PRINT "ner  tambien una  representacion"
 515 PRINT : PRINT "grafica del resultado."
 520 PRINT #0;TAB 8; PAPER 3; INK 4; FLASH 1; BRIGHT 1;"PULS
E UNA TECLA"
 530 PAUSE 0
 540 CLS : PRINT AT 11,8;TAB 8; PAPER 3; INK 5; BRIGHT 1;"FI
N DEL PROGRAMA": PAUSE 0: STOP
 550 REM SUBRUTINA
 560 CLS : PRINT AT 0,12;"CONTROL"
 570 LET P=6
 580 PRINT AT 3,0;"   No.     Valor X    Valor Y"
 590 FOR T=0 TO 31: PRINT "-";: NEXT T
 600 RETURN
```

Lista de variables:

A = Ordenada en el origen de la recta
B = Pendiente de la recta
I = Indice variable
N = Número de pares de valores
S1 =
S2 = } Sumas auxiliares para calcular B y A
S3 =
S4 =
X = Abcisa
Y = Ordenada
Z1 = Campos auxiliares para
Z2 = la introducción de datos

Descripción del programa:

Línea 10-140 : Título, aclaraciones y espera

Línea 150-160 : Indicación del número de pares de valores de la característica y dimensionado

Línea 170-240 : Introducción de los valores de la característica mediante INPUT

Línea 250 : Impresión de la tabla

Línea 260-300 : Impresión de control de los pares de valores y pausa después de 16 pares

Línea 310-320 : Espera

Línea 330-370 : Formación de las cuatro sumas necesarias en las fórmulas para A y para B

Línea 380-410 : Cálculo de los parámetros de la recta A y B

Línea 420-470 : Impresión de los resultados

Línea 480-540 : Indicación para bifurcar y final del programa

---

Línea 550-600 : Subrutina de imprimir la tabla

9.12 Permutaciones

Presentación del problema

El cálculo de permutaciones responde a la cuestión de cuántas formas distintas pueden ordenarse diferentes elementos.

Análisis del problema

Si por ejemplo tenemos tres elementos distintos a, b, y c, pueden distinguirse las siguientes posibilidades de ordenación:

abc, acb, bac, bca, cab, cba

Es decir, que con tres elementos se obtienen seis posibilidades distintas de ordenación.

Generalmente se cumple la siguiente regla:

Para el primero de los tres elementos existen tres posibilidades : puede estar en primer lugar, en segundo lugar o en tercer lugar;

entonces, para el segundo elemento existen solamente dos posibilidades cuando se ha decidido ya dónde debe colocarse el primer elemento.

Esto significa, no obstante, que los dos primeros juntos tienen 3*2=6 posibilidades.

Para el tercer elemento ya sólo queda una posibilidad y para los tres elementos 3*2*1 = 6 posibilidades distintas de ordenación.

De la misma forma, cuando se tienen n elementos distintos, resulta:

Número de posibilidades de ordenación =
n*(n-1)*(n-2)*(n-3)*...*3*2*1

Esta cadena de productos se resume mediante n! (n factorial), y se dice que n elementos pueden ordenarse de n factorial formas distintas, o :

ue la permutación de n elementos distintos es n-factorial (n!).
Debe tenerse en cuenta que para valores de n elevados la expresión n! crece muy rapidamente. Es más, crece tan rápido, que el ordenador ya no puede representar el resultado de 34! a partir de n = 34.

```
  10 REM M-II/11 -PERMUTACIONES-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA DETERMINA
R EL NUME"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"RO DE PERMUTACIONES DE
N ELEMEN-"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"TOS TOMADOS DE N EN N.
       "
  50 PRINT : PRINT "    ------------------------  "
  60 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  70 PRINT "    ------------------------  "
  80 PRINT AT 10,1;"Este programa calcula el facto-"
  90 PRINT : PRINT : PRINT "rial (n!) del valor indicado por
"
  95 PRINT : PRINT : PRINT "el usuario."
 100 INPUT 0: INPUT TAB 2;"NUMERO DE ELEMENTOS n = ";N
 110 LET P=N
 120 IF N>34 OR N<0 THEN  GO TO 100
 130 IF N=0 OR N=1 THEN  LET P=1: GO TO 170
 140 FOR I=N-1 TO 1 STEP -1
 150 LET P=P*I
 160 NEXT I
 170 CLS : PRINT AT 9,3;"EL FACTORIAL DE ";N;" ES : ";AT 12,
12;P
 180 PRINT #0;TAB 6; PAPER 3; INK 6; BRIGHT 1; FLASH 1;"OTRO
 CALCULO (S/N)"
 190 LET A$=INKEY$
 200 IF A$="S" THEN  CLS : GO TO 10
 210 IF A$="N" THEN  GO TO 220
 215 GO TO 190
 220 CLS : PRINT AT 11,8; FLASH 1; INK 3; PAPER 4; BRIGHT 1;
"FIN DEL PROGRAMA": PAUSE 0
```

Lista de variables:

A$ = Variable de cadena para las respuestas

I = Contador

N = Número de elementos

P = Producto intermedio y final

Descripción del programa:

Línea 10-95	: Título y aclaraciones
Línea 100-110	: Demanda de la información INPUT (número de elementos a permutar N) el campo del producto P toma el valor del campo, como primer factor de la cadena de productos a realizar
Línea 120-130	: Comprobación de que no se exceden los límites de cálculo del ordenador.
Línea 140-160	: Formación de la cadena de productos en el campo P
Línea 170	: Impresión del resultado
Línea 180-220	: Final del programa, a menos que se desee continuar calculando (entonces seguir en la línea 10 una vez borrada la pantalla)

9.13 Combinaciones

Presentación del problema

El problema es similar al del programa anterior, calculándose la cantidad de formas distintas en que pueden seleccionarse k elementos de entre n elementos distintos.

La fórmula de cálculo es la siguiente:

$$\text{Número} = \frac{n!}{k!(n-k)!}$$

A esta magnitud se le llama coeficiente binomial y se abrevia de la forma siguiente:

$$\binom{n}{k}$$

Por otra parte, éste equivale a los factores que resultan en el desarrollo de los llamados binomios:

$$(a+b)^2 = a^2 + 2ab + b^2$$

$$= 1a^2b^0 + 2a^1b^1 + 1a^0b^2$$

$$= \binom{2}{0} a^2 b^0 + \binom{2}{1} a^1 b^1 + \binom{2}{2} a^0 b^2$$

$$
\begin{aligned}
(a+b)^3 &= a^3 + 3a^2 b + 3ab^2 + b^3 \\
&= 1a^3 b^0 + 3a^2 b^1 + 3a^1 b^2 + 1a^0 b^3 \\
&= \binom{3}{0} a^3 b^0 + \binom{3}{1} a^2 b^1 + \binom{3}{2} a^1 b^2 + \binom{3}{3} a^0 b^3
\end{aligned}
$$

Como puede observarse, todo binomio se descompone en sumandos. Siempre hay un sumando más que el indicado por el exponente del binomio (el exponente en el último ejemplo era 3; el número de sumandos 4).

Cada sumando consta de 3 factores :

- una constante (el coeficiente binomial que aquí nos ocupa)
- factor "a elevado..." con un exponente, que partiendo del exponente del binomio (aquí 3) baja hasta 0;
- factor b "b elevado a..." con un exponente, que partiendo de 0 sube hasta el exponente del binomio (aquí 3).

En el coeficiente binomial que nos ocupa, el mayor valor es el exponente del binomio (aquí 3); el valor menor es un número entero que va ascendiendo de derecha a izquierda desde 0 hasta el exponente del binomio.

El programa presentado determina cualquier coeficiente binomial, permitiéndonos calcular fácilmente también binomios más complejos (p.e. (a+b)).
Pero este programa puede utilizarse también para cálculos, como se demostrará en los próximos programas.
Finalmente, un programa de este tipo, permite responder p.e. a la siguiente cuestión:

8 corredores están a punto de comenzar una carrera de 100 m lisos. ¿Cuántas posibilidades diferentes existen de reparto de las tres primeras posiciones.?

Respuesta:

$$\binom{8}{3} = \frac{8!}{3!\,(8-3)!} = \frac{8!}{3!\;5!} =$$

$$= \frac{8 \cdot 7 \cdot 6}{3 \cdot 2 \cdot 1} = \underline{\underline{56}}$$

```
  10 REM M-II/12 -COMBINACIONES-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA DETERMINA
R EL NUME"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"RO DE COMBINACIONES DE
N ELEMEN-"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"TOS TOMADOS DE K EN K.
      "
  50 PRINT : PRINT "    ------------------------   "
  60 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  70 PRINT "    ------------------------   "
  80 PRINT AT 11,3;"NUMERO DE ELEMENTOS N : ";: INPUT TAB 12
;"? = ";N: PRINT N
  90 PRINT AT 15,3;"NUMERO SELECCIONADO K : ";: INPUT TAB 12
;"? = ";K: PRINT K
 100 LET Y=N-K
 110 IF K=0 OR K=N THEN  LET BK=1: GO TO 210
 120 IF K>N/2 THEN  LET H=Y: LET Y=K: LET K=H
 130 GO SUB 300
 140 LET PX=A: LET A=N
 150 IF PX=1 THEN  LET PN=N: GO TO 200
 160 FOR I=N-1 TO N-K+1 STEP -1
 170 LET A=A*I
 180 NEXT I
 190 LET PN=A
 200 LET BK=PN/PX
 210 CLS
 220 PRINT AT 1,8; PAPER 1; INK 6; BRIGHT 1;"S O L U C I O N
"
 230 PRINT AT 2,8; INK 3; BRIGHT 1;"---------------"
 240 PRINT AT 7,0;"EL NUMERO DE COMBINACIONES DE ";N
 250 PRINT AT 11,0;"ELEMENTOS TOMADOS DE ";K;" EN ";K;" ES :
"
 260 PRINT AT 15,12;BK
 270 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 5; BRIGHT 1;"PULS
E UNA TECLA"
 280 PAUSE 0
 290 CLS : PRINT AT 11,8; INK 3; PAPER 4; FLASH 1; BRIGHT 1;
"FIN DEL PROGRAMA": PAUSE 0: STOP
 300 REM      SUBRUTINA
 310 LET A=K
 320 IF K=0 OR K=1 THEN  LET A=1: GO SUB 360
 325 IF K<0 THEN  CLS : PRINT AT 11,12;"ERROR": PAUSE 50: GO
 TO 10
 330 FOR I=K-1 TO 1 STEP -1
 340 LET A=A*I
 350 NEXT I
 360 RETURN
```

Lista de variables:

A = Factorial (producto intermedio y final)
BK = Coeficiente binomial
H = Valor auxiliar en el cambio de K por Y
I = Indice variable (factor en el cálculo del factorial)
K = Número de elementos a elegir
N = Número de elementos
PN = Primer factorial en el coeficiente binomial
PX = Segundo factorial en el coeficiente binomial
Y = Y-K

Descripción del programa:

Línea 10-70 : Título y aclaraciones

Línea 80-90 : Introducciones (número de elementos N y número de elementos a elegir K)

Línea 100 : Definición del valor auxiliar Y

Línea 110 : Cálculo simplificado para K=0 ó K=N, porque por deficinición se cumple que

$$\binom{N}{0} = \binom{N}{N} = 1$$

Línea 120 : Intercambio de K y (N-K) (=Y) para K>N/2. Esto es posible porque se cumple que:

$$\binom{N}{K} = \frac{N!}{K!(N-K)!} = \binom{N}{N-K}$$

De este modo se simplifican los cálculos siguientes.

Línea 130 : Cálculo de K! en la subrutina 300

Línea 140-190 : Cálculo de otro factorial, que queda después de reducir (ver p.e. problema de los corredores de antes).

Línea 200-290 : Cálculo e impresión del coeficiente binomial y final del programa principal.

Línea 300-360 : Cálculo de K! en una subrutina

9.14 Lotería

Presentación del problema

Vamos a confeccionar un programa que calcula las posibilidades de premio en la lotería (6 entre 49).

Análisis del problema

El programa calcula la probabilidad de conseguir 3, 4, 5 ó 6 aciertos en la lotería (6 de 49).

Las correspondientes probabilidades se calculan con la siguiente fórmula :

$$P(X) = \frac{\binom{6}{X}\binom{43}{6-X}}{\binom{49}{6}} = \frac{B1*B2}{B3}$$

X es el número de aciertos que debe indicar el usuario.

En suma, aquí tenemos que operar con tres coeficientes binomiales (B1,B2 y B3), que ya conocemos del capítulo anterior.
Por este motivo nos ahorramos ahora cualquier otra explicación.

```
  10 REM M-II/13   -LOTO-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA DETERMINA
R LA PRO-"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"BABILIDAD DE ACERTAR 3,
 4, 5 o 6"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"NUMEROS EN EL LOTO (6 D
E 49).   "
  50 PRINT : PRINT "    ------------------------   "
  60 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  70 PRINT "    ------------------------   "
  80 PRINT AT 11,0;"Este programa no considera la po"
  90 PRINT : PRINT : PRINT "sibilidad de acertar 5 y el nume
"
  95 PRINT : PRINT : PRINT "ro adicional !"
 100 PRINT #0;TAB 8; FLASH 1; PAPER 2; INK 5; BRIGHT 1;"PULS
E UNA TECLA"
 110 PAUSE 0
 120 CLS
 130 PRINT AT 11,5;"NUMERO DE ACIERTOS = ";: INPUT TAB 13;"N
 = ";X: PRINT X
 140 IF X=6 THEN  LET B1=1: LET B2=1: GO TO 400
 150 REM          B1
 160 IF X=5 THEN  LET B1=6: LET B2=43: GO TO 400
 170 LET Y=6-X
 180 LET P=X
 190 FOR I=X-1 TO 1 STEP -1
 200 LET P=P*I
 210 NEXT I
 220 LET P1=P: LET P=Y
 230 FOR I=Y-1 TO 1 STEP -1
 240 LET P=P*I
 250 NEXT I
 260 LET P2=P
 270 LET B1=720/(P1*P2)
 280 REM          B2
 290 LET X=Y
 300 LET P=43
 310 FOR I=P-1 TO 43-X+1 STEP -1
 320 LET P=P*I
 330 NEXT I
 340 LET P1=P
 350 LET P=X
 360 FOR I=X-1 TO 1 STEP -1
```

```
370 LET P=P*I
380 NEXT I
390 LET P2=P: LET B2=P1/P2
400 REM          B3
405 LET P1=49: LET P2=6
410 FOR I=1 TO 5
420 LET P1=P1*(49-I)
430 LET P2=P2*(6-I)
440 NEXT I
450 LET B3=P1/P2: LET W=(B1*B2)/B3
460 CLS
470 IF X=2 THEN  LET X=4
480 PRINT AT 0,7; PAPER 1; INK 6; BRIGHT 1;"R E S U L T A D
O"
490 PRINT AT 1,7; INK 3; BRIGHT 1;"-----------------"
500 PRINT AT 7,0;"LA PROBABILIDAD DE ACERTAR ";X;" NU-"
510 PRINT AT 9,0;"MEROS EN LA LOTERIA PRIMITIVA ES"
520 PRINT AT 12,8;W
530 PRINT #0;TAB 6; PAPER 2; INK 4; FLASH 1; BRIGHT 1;"OTRO
CALCULO (S/N)"
540 LET A$=INKEY$
550 IF A$="S" THEN  CLS : GO TO 10
560 IF A$="N" THEN  GO TO 570
565 GO TO 540
570 CLS : PRINT AT 11,8; PAPER 2; INK 6; BRIGHT 1;"FIN DEL
PROGRAMA": PAUSE 0
```

Lista de variables:

A$ = Variable de cadena para introducir respuestas
B1 = Primer término en la fórmula de cálculo
B2 = Segundo término en la fórmula de cálculo
B3 = Tercer término en la fórmula de cálculo
P = Producto en el cálculo de factoriales
P1 = Primer factorial del coeficiente binomial
P2 = Segundo factorial del coeficiente binomial
W = Probailidad
X = Número de aciertos
Y = 6-X

Descripción del programa:

Línea 10-100 : Título y aclaraciones

Línea 110-120 : Bucle de espera y borrado pantalla

Línea 130 : Introducción del número de aciertos

Línea 140 : Para X=6 se cumple que B1=B2=1 (ver programa Z10); seguir en la línea 400

Línea 150-270 : Determinación de B1 de acuerdo con las reflexiones hechas ya con referencia al programa Z10

Línea 280-390 : Del mismo modo, determinación del segundo coeficiente binomial B2

Línea 400-440 : Lo mismo con B3

Línea 450 : Cálculo de la probabilidad W

Línea 460-520 : Impresión del resultado

Línea 530-570 : Final del programa principal, a menos que se desee un nuevo cálculo

9.15 Distribución binomial

Presentación del problema

La distribución binomial es una distribución de probabilidades. Las distribuciones de probabilidades, a su vez, sirven para calcular -de la forma más sencilla posible- probabilidades, incluso en problemas complejos.

La distribución binomial es también muy importante porque sirve de "base" para toda una serie de procedimientos de cálculo de probabilidades.

La distribución binomial responde a la siguiente cuestión:

¿Cuál es la probabilidad de que en n intentos independientes entre sí se produzca x veces el suceso favorable, si en cada intento sólo son posibles dos resultados, y si la probabilidad de que se produzca este suceso favorable en un intento es del P %.

Sirva el siguiente ejemplo para ilustrar esta cuestión :

¿Cuál es la probabilidad de que echando 10 veces una moneda nos salga p.e. 5 veces cara?

Este es un caso de aplicación de la distribución binomial, pues en cada intento son posibles dos sucesos ("cara" y "cruz") y los intentos (las tiradas) son independientes entre sí.

Otro problema de este tipo sería :

¿Cuál es la probabilidad de que de cada 5 nacimientos 2 sean niña?

Análisis del problema

Para analizar el problema en cuestión nos ocuparemos en primer lugar del ejemplo citado en último lugar :

Si de cada cinco nacimientos dos deben ser niñas, la siguiente ''combinación de nacimientos'' podría responder al problema :

NA NA NO NO NO (NA = niña, NO = niño)

También sería válida la siguiente combinación :

NO NO NO NA NA

y también muchas otras.

El número total de combinaciones "favorables" puede hallarse con ayuda del cálculo de permutaciones, tal como ya comentamos.

Sin ser nuestra intención el demostrar este extremo, calculamos que el número de posibilidades de combinación de 2 niñas y 3 niños es:

$$\text{Número} = \frac{5!}{2! * 3!} = \frac{5*4*3*2*1}{2*1 * 3*2*1}$$

$$= \frac{5*4}{2*1} = 10$$

Generalmente, el número de disposiciones distintas de n elementos, de los que x son iguales (en el sentido de "favorables") y otros n-x son también iguales (en el sentido de "desfavorables"), es :

$$\text{Número} = \frac{n!}{x! * (n-x)!}$$

Para seguir con el análisis del problema nos basaremos en que la probabilidad de que nazca una niña es P = 0.5 - esta es la probabilidad de que se produzca un suceso favorable en un intento.

Contemplemos el resultado siguiente :

NA NA NO NO NO

¿Cuál es la probabilidad de que se produzca un resultado así en cinco intentos?

Contemplemos en primer lugar el nacimiento de una niña : La probabilidad de que se produzca es lógicamente 0.5.

La probabilidad de que los dos primeros nacimientos juntos sean niña es entonces 0.5 * 0.5 (en general P * P).

La probabilidad de que nazca un niño es 1 - 0.5 (en general 1 - P). De esta manera, siguiendo el mismo procedimiento de arriba, la probabilidad de que el tercero, el cuarto y el quinto nacimiento sean niño es :

$$(1 - 0.5) * (1 - 0.5) * (1 - 0.5)$$

o, en general :

$$(1 - P) * (1 - P) * (1 - P)$$

Para la combinación NA NA NO NO NO obtenemos en total la probabilidad siguiente (escrita de forma general) :

$$P * P * P (1 - P) * (1 - P) * (1 - P)$$

$$= P^2 * (1 - P)^3$$

$$= P^{x_i} * (1 - P)^{n-x_i}$$

En nuestro ejemplo obtenemos :

Probabilidad para NA NA NO NO NO =

$$(0.5)^2 * (1 - 0.5)^{5-2} = (0.5)^5 = 1/32$$

Pero, en el ejemplo que nos ocupa hay 10 posibles combinaciones distintas NA,NO .

Así a partir de 10 posibles combinaciones, con una probabilidad de 1/32 cada una, se obtiene :

Probabilidad de que de 5 nacimientos 2

sean niña = 10 * 1/32 = 10/32 = 31.25 %

Este ejemplo nos permite inducir la siguiente fórmula general :

La probabilidad de que en n intentos se produzca x veces el suceso favorable, cuando la probabilidad individual de producirse un suceso es P, se calcula del modo siguiente :

$$W = \frac{n!}{x_i! * (n-x_i)!} * P^{x_i} * (1-P)^{n-x_i}$$

Esta fórmula parece complicada, pero su programación resulta relativamente sencilla.

```
  10 REM M-II/14   -BINOMIAL-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA PARA CALCULAR
LAS PROBA"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"BILIDADES CON LA DISTRI
BUCION BI"
  45 PRINT  PAPER 4; INK 0; FLASH 1;"NOMIAL.
      "
  50 PRINT : PRINT "    -----------------------  "
  60 PRINT AT 5,4; PAPER 1;"PROF. DR. W. VOSS,  1984"
  70 PRINT "    -----------------------  "
  75 INK 5: BRIGHT 1: PRINT AT 9,0;"CON ESTE PROGRAMA, ES DE
CIR, CON"
  80 PRINT : PRINT "LA DISTRIBUCION BINOMIAL,  PUEDE"
  90 PRINT : PRINT "DIRIMIRSE LA SIGUIENTE CUESTION:"
 100 INK 3: PRINT AT 16,0;" Cual es  la probabilidad de que"
 110 PRINT "se produzca x veces un suceso fa"
 120 PRINT "vorable en N intentos, si la pro"
 125 PRINT "babilidad en un intento es P ?"
 130 PRINT #0;TAB 8; PAPER 2; INK 4; FLASH 1;"PULSE UNA TECL
A"
 140 PAUSE 0: INK 7
 150 CLS
 160 PRINT AT 1,6;"DATOS DEL PROBLEMA"
 170 PRINT AT 2,6; INK 2;"------------------"
 180 PRINT AT 7,0;"NUMERO DE INTENTOS..........= ";: INPUT T
AB 12;"N = ";N: PRINT N
 190 PRINT AT 11,0;"NUMERO DE SUCESOS FAVORABLES= ";: INPUT
TAB 12;"X = ";X: PRINT X
 200 PRINT AT 15,0;"PROBABILIDAD EN UN INTENTO..= "
 210 PRINT AT 16,3;"(como numero decimal)": INPUT TAB 12;"P
= ";P
 220 LET Z=X
 230 REM  COEFICIENTE BINOMIAL
 240 LET Y=N-X: LET Q=1-P
 250 IF X=0 OR X=N THEN  LET BK=1: GO TO 350
 260 IF X>N/2 THEN  LET H=Y: LET Y=X: LET X=H: LET G=Q: LET
Q=P: LET P=G: LET F=X
 270 GO SUB 470
 280 LET PX=A: LET A=N
 290 IF PX=1 THEN  LET PN=N: GO TO 340
 300 FOR I=N-1 TO N-X+1 STEP -1
 310 LET A=A*I
 320 NEXT I
```

```
 330 LET PN=A
 340 LET BK=PN/PX
 350 LET W=BK*P^X*Q^Y
 360 CLS
 370 PRINT AT 2,8;"R E S U L T A D O"
 380 PRINT AT 3,8; INK 2;"-----------------"
 390 PRINT AT 7,0;"LA PROBABILIDAD DE PRODUCIRSE EN"
 400 PRINT : PRINT : PRINT : PRINT N;" INTENTOS ";Z;" VECES
EL SUCESO FA"
 410 PRINT : PRINT : PRINT : PRINT "VORABLE ES : ";W
 420 PRINT #0;TAB 7; PAPER 2; INK 4; FLASH 1;"OTRO CALCULO (
S/N)"
 430 LET A$=INKEY$
 440 IF A$="S" THEN  GO TO 150
 450 IF A$="N" THEN  GO TO 460
 455 GO TO 430
 460 CLS : PRINT AT 11,8; FLASH 1; INK 3; PAPER 5;"FIN DEL P
ROGRAMA": PAUSE 0: STOP
 470 REM       SUBRUTINA
 480 LET A=X
 490 IF X=0 OR X=1 THEN  LET A=1: GO TO 530
 500 FOR I=X-1 TO 1 STEP -1
 510 LET A=A*I
 520 NEXT I
 530 RETURN
```

Lista de variables:

A = Producto en el cálculo de factoriales
A$ = Variable de cadena para la introducción de respuestas
BK = Coeficiente binomial
F = Valor auxiliar (F=X)
G = Valor auxiliar en el cambio de P y Q
H = Valor auxiliar en el cambio de X e Y
I = Indice variable
N = Número de intentos
P = Probabilidad de producirse el suceso favorable en cada intento

PN = Primer término del coeficiente binomial
PX = Segundo término del coeficiente binomial
Q = 1-P
W = Probabilidad
X = Número de sucesos favorables
Y = N-X
Z = X

Descripción del programa

Línea 10-125 : Título y aclaraciones

Línea 130-210 : Espera e introducción de las informaciones INPUT (número de intentos, probabilidad).

Línea 220-340 : Determinación del coeficiente binomial :

240 : Campos auxiliares;
250 : Si X=0 ó X=1 entonces el coeficiente se define como 1; seguir en 350;
260 : Si X>N/2 entonces, para aprovechar las posibilidades de reducción, cambiar X por N-X y P por 1-P
280 : Definición del denominador (=PX) del coeficiente y redefinición de A,en caso de utilizar nuevamente var. A

290 : Si el denominador del coeficiente es = 1, entonces el numerador queda definido como N; seguir en línea 340.
300-320 : Determinación del numerador del coeficiente BK;
340 : Cálculo de BK.

Línea 350 : Cálculo de la probabilidad buscada.

Línea 360-410 : Impresión del resultado del cálculo.

Línea 420-460 : Final del programa, a menos que no se quiera un nuevo cálculo (entonces volver a la línea

---

Línea 470-530 : Subrutina para calcular el factorial

9.16 Resolución de sistemas de ecuaciones

Presentación del programa

Cuando nos encontramos con problemas cuya resolución requiere el cálculo de un sistema de N ecuaciones con N incógnitas cada una, es conveniente tener un programa que nos permita librarnos de los tediosos y repetitivos cálculos a que nos llevaría un sistema de gran envergadura.

El algoritmo del programa se basa en aplicar el método de Gauss-Jordan a la matriz ampliada del sistema. Este método viene explicado en cualquier tratado de álgebra lineal existente en el mercado por lo que nos extenderemos en su explicación.

El programa pregunta en primer lugar la dimensión de la matriz ampliada del sistema que tenemos que resolver. A continuación se nos van pidiendo los coeficientes de cada ecuación (si falta alguna incógnita deberemos introducir un cero).
Una vez introducidos todos los coeficientes se pasa al cálculo de las soluciones.

Este programa no tiene en cuenta los posibles casos de incompatibilidad; instamos al lector a que modifique el programa teniendo en cuenta estos posibles casos.

```
  10 REM M-II/15   -MATRICES-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"ESTE PROGRAMA RESUELVE
 SISTEMAS"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"DE ECUACIONES MEDIANTE
LA APLICA"
  50 PRINT  PAPER 4; INK 0; FLASH 1;"CION DEL METODO DE GAUS
S-JORDAN."
  60 PRINT AT 15,4; PAPER 3; INK 5; BRIGHT 1; FLASH 1;"Pulse
 una tecla por favor"
  70 PAUSE 0
  80 CLS
  90 PRINT AT 11,0;"DIMENSION DE LA MATRIZ DE COEFI-": PRINT
 : PRINT "       CIENTES DEL SISTEMA?": INPUT TAB 12;"N = ";N
 100 DIM A(N+1,N+1)
 110 CLS
 120 FOR I=1 TO N
 130 PRINT "COEFICIENTES DE LA ";I;"a";" ECUACION": PRINT :
PRINT : PRINT : PRINT
 140 FOR J=1 TO N
 150 PRINT "    A(";I;",";J;")    =  ";: INPUT TAB 10;A(I,J)
: PRINT A(I,J): PRINT : PRINT
 160 NEXT J
 170 INPUT "    Termino cte. b = ";A(I,N+1): CLS
 180 NEXT I
 190 CLS : PRINT AT 12,7; FLASH 1; PAPER 2; INK 5; BRIGHT 1;
"C A L C U L A N D O"
 200 FOR I=1 TO N-1
 210 FOR J=I+1 TO N+1
 220 LET IN=A(J,I)/A(I,I)
 230 FOR X=1 TO N+1
 240 LET A(J,X)=A(J,X)-IN*A(I,X)
 250 NEXT X
 260 NEXT J
 270 NEXT I
 280 FOR I=1 TO N
 290 IF A(I,I)=1 THEN  GO TO 340
 300 LET IN=1/A(I,I)
 310 FOR X=1 TO N+1
 320 LET A(I,X)=IN*A(I,X)
 330 NEXT X
 340 NEXT I
 350 FOR I=N TO 1 STEP -1
 360 LET SU=0
```

```
 370 FOR X=I+1 TO N
 380 LET SU=SU+A(X,X)*A(I,X)
 390 NEXT X
 400 LET A(I,I)=A(I,N+1)-SU
 410 NEXT I
 420 CLS
 430 PRINT AT 1,4;"LAS SOLUCIONES SON :": PRINT : PRINT : PR
INT
 440 FOR I=1 TO N
 450 IF A(I,I)-INT (A(I,I))>.99999 THEN  LET W=A(I,I)+.00001
: LET W=INT ((W*10000+.5)/10000): PRINT "     X";I;" =";" ";
W: PRINT : GO TO 470
 460 PRINT "     X";I;" =";" ";A(I,I): PRINT
 470 NEXT I
 480 IF N<9 THEN  PRINT #0;TAB 8; FLASH 1; INK 2; PAPER 4;"F
IN DEL PROGRAMA": PAUSE 0: STOP
 485 PRINT #0;TAB 8; FLASH 1; INK 2; PAPER 4;"FIN DEL PROGRA
MA"
 490 PAUSE 0: STOP
```

Descripción del programa:

Línea 10-50 : Título y descripción del programa

Línea 60-70 : Bucle de espera.

Línea 80 : Borrado de la pantalla

Línea 90 : Petición de la dimensión de la matriz de coeficientes del sistema.

Línea 100 : Dimensionado de la matriz A con las dimensiones del sistema.

Línea 110-180 : Petición de los coeficientes y término constante de cada ecuación

Línea 190 : Impresión del mensaje ''CALCULANDO''

Línea 200-410 : Cálculo de la diagonalización de la matriz ampliada y de las soluciones del sistema.

Línea 420-490 : Impresión de las soluciones

9.17 Elecciones

Presentación del problema

El programa representa gráficamente, en forma de un diagrama de barras, los resultados de las elecciones.

Para ello, el usuario debe introducir el porcentaje de votos obtenido por los partidos señalados en el programa : PSOE, AP, CIU, PCE y otros.

Análisis del problema

Dada la sencillez del problema podemos prescindir del análisis del mismo.

```
  10 REM M-II/16  -ELECCIONES-
  20 BORDER 0: PAPER 0: INK 7: CLS
  25 POKE 23658,8: POKE 23609,35
  30 PRINT  PAPER 4; INK 0; FLASH 1;"PROGRAMA  PARA LA REPRE
SENTACION"
  40 PRINT  PAPER 4; INK 0; FLASH 1;"GRAFICA DE  RESULTADOS
ELECTORA-"
  50 PRINT  PAPER 4; INK 0; FLASH 1;"LES.
     "
  60 PRINT "         -----------------"
  70 PRINT AT 4,8; PAPER 1; INK 4; BRIGHT 1;"PROF. DR. W. VO
SS"
  80 PRINT "         -----------------"
  84 DIM F(5): DIM G(5)
  86 PRINT AT 8,0;"DEME LOS PORCENTAJES ELECTORALES"
  90 PRINT AT 11,10;"PSOE..... = ";: INPUT TAB 12;"? = ";F(1
): PRINT F(1)
 100 PRINT AT 13,10;"CP....... = ";: INPUT TAB 12;"? = ";F(2
): PRINT F(2)
 110 PRINT AT 15,10;"CDC...... = ";: INPUT TAB 12;"? = ";F(3
): PRINT F(3)
 120 PRINT AT 17,10;"PCE...... = ";: INPUT TAB 12;"? = ";F(4
): PRINT F(4)
 130 PRINT AT 19,10;"OTROS.... = ";: INPUT TAB 12;"? = ";F(5
): PRINT F(5)
 140 LET S=0
 150 FOR I=1 TO 5
 160 LET S=S+F(I)
 170 NEXT I
 180 IF S=100 THEN  GO TO 190
 185 CLS : PRINT AT 11,0;"ERROR EN LA INTRODUCCION DE POR-";
AT 13,10;"CENTAJES.": PRINT #0;TAB 8;"PULSE UNA TECLA": PAUS
E 0: CLS : GO TO 86
 190 CLS
 200 LET FM=0
 210 FOR I=1 TO 5
 220 IF F(I)>FM THEN  LET FM=F(I)
 230 NEXT I
 240 FOR I=1 TO 5
 250 LET G(I)=(13/FM)*F(I)
 260 LET G(I)=INT (G(I)+.5)
 270 NEXT I
 280 PRINT AT 0,6; PAPER 1; INK 5; BRIGHT 1; FLASH 1;"R E S
U L T A D O S"
 290 PRINT  INK 6; BRIGHT 1;"       -------------------"
```

```
 300 PRINT AT 7,9;"PSOE..... = ";F(1)
 310 PRINT AT 9,9;"CP....... = ";F(2)
 320 PRINT AT 11,9;"CDC...... = ";F(3)
 330 PRINT AT 13,9;"PCE...... = ";F(4)
 340 PRINT AT 15,9;"OTROS.... = ";F(5)
 350 PRINT #0;TAB 8; FLASH 1; PAPER 3; INK 4; BRIGHT 1;"PULS
E UNA TECLA"
 360 PAUSE 0: CLS
 370 REM      GRAFICO
 380 FOR J=1 TO 5
 390 LET B=17: LET A=17-G(J)
 400 FOR I=A TO B
 410 PRINT AT I,(J-1)*6+2;"*"
 420 PRINT AT I,(J-1)*6+3;"*"
 430 PRINT AT I,(J-1)*6+4;"*"
 440 NEXT I
 450 NEXT J
 460 FOR I=0 TO 31
 470 PRINT AT 17,I;"#"
 480 NEXT I
 490 PRINT AT 19,0;" PSOE    CP   CDC   PCE  OTROS"
 500 PRINT #0;TAB 8; FLASH 1; INK 2; PAPER 4;"FIN DEL PROGRA
MA": PAUSE 0
```

Lista de variables:

A = Línea de inicio del gráfico

B = Línea final del gráfico

F = Porcentajes

FM = Valor máximo

G = Porcentajes transformados

I = Indice variable

J = Indice variable

S = Campo para sumas

X = Campo para introducir datos

Descripción del programa:

Línea 10-80	: Título y aclaraciones
Línea 84	: Dimensionados
Línea 86-130	: Introducción de las informaciones INPUT
Línea 140-180	: Prueba
Línea 190	: Borrado de la pantalla
Línea 200-230	: Búsqueda del porcentaje mayor
Línea 240-270	: Conversión de los porcentajes de manera que el valor máximo aproveche el espacio disponible en la pantalla
Línea 280-360	: Impresión de las informaciones
Línea 370-450	: Impresión del gráfico
Línea 460-480	: Impresión de una línea horizontal
Línea 490	: Impresión de los textos PSOE, AP, CDC, PCE, OTROS.
Línea 500	: Final del programa.

Epílogo

En los capítulos precedentes hemos intentado mostrar cómo pueden resolverse problemas típicos de la escuela con programas BASIC relativamente sencillos.

Desearía insistir en que los ejemplos escogidos tienen únicamente carácter de muestra, y que no se le ha dado concedido especial importancia a los programas óptimos.

El lector debe comprender rápidamente el funcionamiento de los programas. De este modo será capaz de encontrar la vía de resolución de otros problemas, que frecuentemente se asemejarán a los que aquí presentamos.

Lo fundamental a la hora de utilizar un ordenador no es el ordenador en sí, sino la capacidad que demuestra el usuario para tratar determinados problemas, de forma que pueda ofrecerle al ordenador el tipo de resolución más adecuada para un programa.

No es necesario que un programa funcione perfectamente a la primera (aun cuando esto, naturalmente, sea muy positivo); el usuario debe aprender cómo estructurar mentalmente un problema dado, para que el ordenador pueda resolverlo :

La causa es que el ordenador no sirve para nada si nosotros, los usuarios, no conocemos la vía de resolución con anterioridad.

Si alguna vez ocurriese que un programa no funciona tal y como nosotros lo habíamos previsto, informándonos mediante un mensaje de error y la interrupción del programa, siempre nos ayudará el echar una ojeada al manual del ordenador.

A aquellos lectores, a los que los ejemplos presentados les han parecido demasiado sencillos, les indicaremos que este libro ha sido concebido en primera instancia para principiantes del BASIC y que no hemos querido superar sus posibilidades a través de una complejidad excesiva de la obra. Para consuelo de los primeros, queremos anunciar la pronta publicación de un segundo tomo de esta obra, donde se estudiarán problemas más complejos, especialmente de tipo matemático.

A

B

C

D

E

F

G

H

L

M

N

O

P

P

Q

R

R

S

T

U

V

V

Z



ROBOTICA PARA SU COMMODORE 64, 230 pág.
P.V.P. 2.800,- ptas.

En el libro de los robots se muestran las asombrosas posibilidades que ofrece el CBM 64, para el control y la programación, presentadas con numerosas ilustraciones e intuitivos ejemplos. El punto principal: Cómo puede construirse uno mismo un robot sin grandes gastos. Además, un resumen del desarrollo histórico del robot y una amplia introducción a los fundamentos cibernéticos.
Gobierno del motor, el modelo de simulación, interruptor de pantalla, el Port-Usuario cómodo del modelo de simulación, Sensor de infrarrojos, concepto básico de un robot, realimentación unidad cibernética, Brazo prensor, Oír y ver.



MANUAL ESCOLAR PARA SU COMMODORE 64, 351 pág.
P.V.P. 2.800,- ptas.

Este libro, escrito especialmente para escolares de grado medio y superior, contiene muchos interesantes programas de aprendizaje para solucionar problemas, descritos detalladamente y de manera fácilmente comprensible. Facilitan un aprendizaje intensivo y ameno, con, entre otros, los siguientes temas: Teorema de pitágoras, progresiones geométricas, palanca mecánica, crecimiento exponencial, verbos irregulares, ecuaciones de segundo grado, movimientos de péndulo, formación de moléculas, aprendizaje de vocablos, cálculo de interés y su capitalización. Una corta repetición de los elementos BASIC más importantes y una introducción a los rasgos esenciales del análisis de problemas, entre otros, completan el conjunto.



PEEKS y POKES, 177 pág.
P.V.P. 1.600,- ptas.

Con importantes comandos PEEK y POKE se pueden hacer también desde el Basic muchas cosas, para las que se necesitarían normalmente complejas rutinas en lenguaje máquina. Este libro explica de manera sencilla el manejo de PEEKs y POKEs. Con una enorme cantidad de POKEs importantes y su posible aplicación. Para ello se explica perfectamente la estructura del Commodore 64: Sistema operativo, interpretador, página cero, apuntadores y stacks, generador de caracteres, registros de sprites, programación de interfaces, desactivación del interrupt. Además una introducción al lenguaje máquina. Muchos programas ejemplo.



TODO SOBRE EL FLOPPY 1541,
P.V.P. 3.200,- ptas.

La obra Standard del floppy 1541, todo sobre la programación en disquettes desde los principiantes a los profesionales, además de las informaciones fundamentales para el DOS, los comandos de sistema y mensajes de error, hay varios capítulos para la administración práctica de ficheros con el FLOPPY, amplio y documentado Listado del Dos. Además un filón de los más diversos programas y rutinas auxiliares, que hacen del libro una lectura obligada para los usuarios del Floppy.



MANTENIMIENTO Y REPARACION DEL FLOPPY 1541,
200 pág.
P.V.P. 2.800,- ptas.

Saberse apañar uno mismo, ahorra tiempo, molestias y dinero, precisamente problemas como el ajuste del floppy o reparaciones de la platina se pueden arreglar a menudo con medios sencillos. Instrucciones para eliminar la mayoría de perturbaciones, listas de piezas de recambio y una introducción a la mecánica y a la electrónica de la unidad de disco, hay también indicaciones exactas sobre herramientas y material de trabajo. Este libro hay que considerarlo en todos sus aspectos como efectivo y barato.



EL MANUAL DEL CASSETTE, 190 pág.
P.V.P. 1.600,- ptas.

Un excelente libro, que le mostrará todas las posibilidades que le ofrece su grabadora de casettes. Describe detalladamente, y de forma comprensible, todo sobre el Datassette y la grabación en cassette. Con verdaderos programas fuera de serie: Autostart, Catálogo (¡busca y carga automáticamente!), backup de y a disco, SAVE de áreas de memoria, y lo más sorprendente: un nuevo sistema operativo de cassette con el 10-20 veces más rápido FastTape. Además otras indicaciones y programas de utilidad (ajuste de cabezales, altavoz de control).



MSX EL MANUAL ESCOLAR
P.V.P. 2.800,- ptas.

Escrito para alumnos de los últimos cursos de EGB y de BUP, este libro contiene muchos programas para resolver problemas y de aprendizaje, descritos de una forma muy completa y fácil de comprender. Teorema de Pitágoras, progresiones geométricas, escritura cifrada, crecimiento exponencial, verbos irregulares, igualdades cuadráticas, movimiento pendular, estructura de moléculas, cálculo de interés y muchas cosas más.



MSX GRAFICOS Y SONIDOS, 250 pág.
P.V.P. 2.800,- ptas.

Las computadoras MSX no sólo ofrecen una relación precio/rendimiento sobresaliente, sino que también poseen unas cualidades gráficas y de sonido excepcionales. Este libro expone las posibilidades de los MSX de forma completa y fácil. El texto se completa con numerosos y útiles programas ejemplo.



MSX PROGRAMAS Y UTILIDADES, 1985, 194 pág.
P.V.P. 2.200,- ptas.

El libro contiene una amplia colección de importantes programas que abarcan, desde un desensamblador hasta un programa de clasificaciones deportivas. Juegos superemocionantes y aplicaciones completas. Los programas muestran además importantes consejos y trucos para la programación. Estos programas funcionan en todos los ordenadores MSX, así como en el SPECTROVIDEO 318 328. ESTRACTO DEL CONTENIDO: Volcado memoria hexadecimal. Editor gráficos. Editor de sonido. Escritura de ordenador. Lista referencia de variables. Calendario. Desensamblador. ADMINISTRACION de una colección de discos L.P. HOLLOW - JUEGO DE LAS CEREZAS. DIAGRAMAS DE BARRAS. TABLAS DEPORTIVAS.



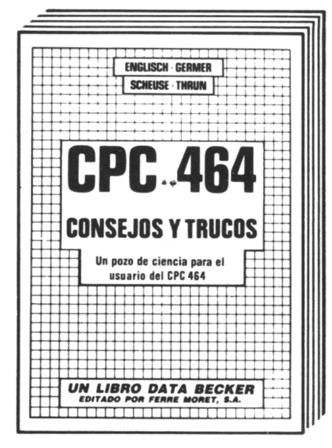
64 EN EL CAMPO DE LA TECNICA Y LA CIENCIA, 296 pág.
P.V.P. 2.800,- ptas.

Ofrece un campo fascinante y amplio de problemáticas científicas. Para esto el libro contiene muchos listados interesantes: Análisis de Fornier y síntesis, análisis de redes, exactitud de cálculo, formateado de números, cálculo del valor PH, sistemas de ecuaciones diferenciales, modelo ladrón presa, cálculo de probabilidad, medición de tiempo, integración, etc.



CPC-464 EL LIBRO DEL COLEGIO
P.V.P. 2 800,- ptas.

Escrito para alumnos de los últimos cursos de EGB y de BUP, este libro contiene muchos programas para resolver problemas y de aprendizaje, descritos de una forma muy compleja y fácil de comprender. Teorema de Pitágoras, progresiones geométricas, escritura cifrada, crecimiento exponencial, verbos irregulares, igualdades cuadráticas, movimiento pendular, estructura de moléculas, cálculo de interés y muchas cosas más.



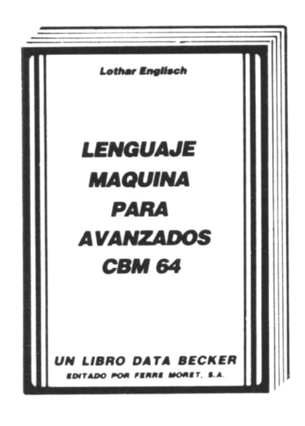
CPC-464 CONSEJOS Y TRUCOS
P.V.P. 2.200,- ptas.

Ofrece una colección muy interesante de sugerencias, ideas y soluciones para la programación y utilización de su CPC-464: Desde la estructura del hardware, sistema de funcionamiento - Tokens Basic, dibujos con el joystick, aplicaciones de ventanas en pantalla y otros muchos interesantes programas como el procesamiento de datos, editor de sonidos, generador de caracteres, monitor de código máquina hasta listados de interesantes juegos.



LENGUAJE MAQUINA PARA
COMMODORE 64, 1984, 201 pág.
P.V.P. 2.200,- ptas.

¡Por fin una introducción al código máquina fácilmente comprensible! Estructura y funcionamiento del procesador 6510, introducción y ejecución de programas en lenguaje máquina, manejo del ensamblador, y un atractivo muy especial: ¡un simulador de paso a paso escrito en BASIC!



LENGUAJE MAQUINA PARA AVANZADOS
CBM 64, 1984, 206 pág.
P.V.P. 2.200 ptas.

¿Ud. ha logrado iniciarse en código máquina? Entonces el «nuevo English» le enseñará cómo convertirse en un profesional. Naturalmente con muchos programas ejemplo, rutinas completas en código máquina e importantes consejos y trucos para la programación en lenguaje máquina y para el trabajo con el sistema operativo.



EL ENSAMBLADOR

Este libro ofrece al programador interesado una introducción fácilmente comprensible para los tan extendidos Assembler PROFI-ASS, SM MAE y T.EX.ASS. con consejos y trucos de gran utilidad, indicaciones y programas adicionales. Al mismo tiempo sirve de manual orientado a la práctica, con aclaraciones de conceptos importantes e instrucciones.
250 páginas. 2.200,- ptas.



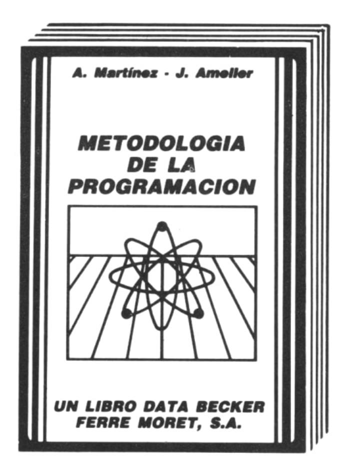
ZX SPECTRUM EL MANUAL ESCOLAR
P.V.P. 2.200,- ptas.

Escrito para alumnos de los últimos cursos de EGB y de BUP, este libro contiene muchos programas para resolver problemas y de aprendizaje, descritos de una forma muy completa y fácil de comprender. Teorema de Pitágoras, progresiones geométricas, escritura cifrada, crecimiento exponencial, verbos irregulares, igualdades cudráticas, movimiento pendular, estructura de moléculas, cálculo de interés y muchas cosas más.



ZX SPECTRUM CONSEJOS Y TRUCOS, 211 pág.
P.V.P. 2.200,- ptas.

Una interesante colección se sugestivas ideas y soluciones para la programación y utilización de su ZX ESPECTRUM. Aparte de muchos peeks, pokes y USRs hay también capítulos completos para, entre otros, entrada de datos asegurado sin bloqueo de ordenador, posibilidades de conexión y utilización de microdrives y lápices ópticos programas para la representación de diagramas de barra y de tarta, el modo de utilizar óptimamente ROM y RAM.



METODOLOGIA DE LA PROGRAMACION
P.V.P. 2.200,- ptas.

El primer libro recomendado para escuelas de enseñanza de informática y para aquellas personas que quieren aprender la programación. Cubre las especificaciones del Ministerio de Educación y Ciencia para Estudios de Informática. Realizado por un alto mando del ejército Español, un Dr. Ingeniero y Diplomado en Informática y profesor de la UNED y por un oficial técnico especialista en informática de gestión. Utilizado en todos los institutos politécnicos del ejército español. Es un seguro candidato a ediciones en lengua inglesa, alemana y francesa. Es el primer libro que introduce a la lógica del ordenador. Es un elemento de base que sirve como introducción para la programación en cualquier otro lenguaje. No se requieren conocimientos de programación ni siquiera de informática. Abarca desde los métodos de programación clásicos a los más modernos.



TODO SOBRE EL NUEVO COMMODORE 128

P.V.P. 2.200,- ptas.

El Libro de Primicias del Commodore 128 no ofrece solamente un resumen completo de todas las características y rendimientos del sucesor del C-64 y con ello una importante ayuda para su adquisición. Muestra, además, todas las posibilidades del nuevo equipo en función de sus tres modos de operación.

Entre otros se describen el hardware, los modos de operación: modo 64, modo 128 y modo CP/M, las configuraciones de memoria, la disposición de la página cero, trabajos con dos pantallas, modo de 80 caracteres, Basic V 7.0: comandos de gráficos y sonidos, comandos de control, periféricos rápidos (1571) etc.

EL CONTENIDO:

Escrito para alumnos de los últimos cursos de EGB y de BUP, este libro contiene multitud de programas para resolver problemas y de aprendizaje, descritos de una forma muy completa y fácil de comprender. Proporcionan un aprendizaje intensivo, a la vez que divertido, con el ZX SPECTRUM.

Del contenido:

— Teorema de Pitágoras
— Progresiones geométricas
— Escritura cifrada
— Crecimiento exponencial
— Verbos irregulares
— Igualdades cuadráticas
— Movimiento pendular
— Estructura de moléculas
— Repaso de vocablos
— Cálculo de intereses

Una pequeña panorámica sobre los fundamentos del tratamiento informático, un somero repaso a los principales elementos del BASIC y una introducción en los aspectos básicos del análisis de problemas completan el libro.

ESTE LIBRO HA SIDO ESCRITO POR:

Werner Voss, profesor de estadística en la Universidad de Bochum. Multitud de publicaciones dentro del campo de la estadística y del tratamiento informático.

